

Automatic prior shape selection for image edge detection with modified Mumford–Shah model

Yuying Shi ^{a,*}, Zhimei Huo ^a, Jing Qin ^b, Yilin Li ^c

^a Department of Mathematics and Physics, North China Electric Power University, Beijing 102206, PR China

^b Department of Mathematics, University of Kentucky, Lexington, KY 40506, USA

^c Beijing ViSystem Corporation Limited, Beijing, PR China

ARTICLE INFO

Article history:

Received 3 December 2018

Received in revised form 3 July 2019

Accepted 23 September 2019

Available online 3 October 2019

Keywords:

Edge detection

Prior shape

ADMM

Fixed-point iterative algorithm

Automatic shape selection

Modified Mumford–Shah model

ABSTRACT

Edge detection plays an important role in the field of image processing. In this paper, we propose a novel variational model to automatically and adaptively detect one or more prior shapes from the given dictionary to guide the edge detection process. In that way, we can effectively detect the shapes of interest from the test image. Moreover, an efficient algorithm based on the Alternating Direction Method of Multipliers (ADMM) is proposed to solve this model with guaranteed convergence. A variety of numerical experiments show that the proposed method has achieved ideal performance for edge detection in images with missing information, various types of noise and complicated background, and even multiple objects.

© 2019 Elsevier Ltd. All rights reserved.

1. Introduction

Edge detection techniques aim to extract the edges of objects within an image. It has been applied in various fields, including computer vision, machine learning, and medical imaging [1,2]. Solving the edge detection problem in a reliable way has been of prime interest for several decades and has produced an enormous number of publications.

There has been a large amount of operator-based methods developed for edge detection. For example, gradient operators include Sobel operator [3], Roberts operator [4], Prewitt operator [5], and second-order differential operators include Canny operator [6], and LOG operator [7]. However, these methods may lead to the loss of some geometric features and corruption of edges in the original image because of gradient discretization. To address this issue, some methods have been proposed to improve the accuracy of detected edges, including fuzzy algorithms [8], adaptive splitting algorithm [9], wavelets [10], G-lens [11], and anisotropic diffusions [12–14]. In [15], a comprehensive comparison of various edge detection methods for various image processing applications is provided. It is worth noting that edge detection becomes very challenging when the original image has no salient edges or it is corrupted by non-Gaussian noise or a mixture of several types of noise.

Another popular category of edge detection methods is to solve a minimization model which describes noise-reduction and edge-detection properly and explicitly. One representative work is the popular Mumford–Shah (MS) model [16]:

$$\min_{u, \Gamma} \left\{ E(u, \Gamma) = \mu \int_{\Omega \setminus \Gamma} |\nabla u|^2 d\mathbf{x} + \frac{\nu}{2} \int_{\Omega} (u - I)^2 d\mathbf{x} + |\Gamma| \right\}, \quad (1)$$

* Corresponding author.

E-mail addresses: yyshi@amss.ac.cn (Y. Shi), huozhm@163.com (Z. Huo), jing.qin@uky.edu (J. Qin), yllncepu@163.com (Y. Li).

where I is a given image on an open bounded domain $\Omega \subset \mathbb{R}^2$, the minimizer u is expected to be a “good” piecewise smooth approximation of I , and $|\Gamma|$ is the total length of the union of the targeted edges Γ . Here μ and ν are positive tuning parameters. It is difficult to solve the model explicitly because of the third term about the edge length. Instead, many methods have been proposed to solve an approximate model of (1) which is easier to obtain the minimizer. In particular, Ambrosio and Tortorelli proposed a Γ -convergence approximation via the uniformly elliptic functionals [17,18]. To overcome the over-segmentation of steep gradients, Blake and Zisserman proposed a variational model based on second-order derivatives [19]. Later, Zanetti et al. combined the above two models to enhance the detection effectiveness [20]. To detect open curves accurately, Wang et al. proposed the modified Mumford–Shah (MMS) model by integrating the MS model with the binary level-set method [21], which is only effective for the Gaussian type of noise. Recently, Shi et al. replaced the L_2 fidelity term by the L_1 fidelity term to enhance robustness to other noise types, e.g., impulse noise [22]. To overcome the ill-posedness of the MMS model [23,24], Shi et al. has recently improved the MMS model by adding a L_1 regularization term of the edge and proposed the following model [25]:

$$\min_{u, \psi \in [0,1]} \left\{ \mu \int_{\Omega} (1 - \psi)^2 |\nabla u|^2 d\mathbf{x} + \frac{\nu}{2} \int_{\Omega} (u - I)^2 d\mathbf{x} + TV(\psi) + \tau \int_{\Omega} |\psi| d\mathbf{x} \right\}. \quad (2)$$

The minimizer u belongs to the space of functions defined on Ω with bounded variations, denoted by $BV(\Omega)$, and u is expected to be a piecewise smooth approximation of I . The tuning parameters μ, ν, τ are positive. The binary level set function $\psi = 1$ if $\mathbf{x} \in R_d$, while $\psi = 0$ otherwise. The closed region R_d is generated by a regular current and its parallel or offset curve r_d (refer to [25] for the concrete definition of R_d). Experimental results show that the model (2) can identify clear edges for both open and closed edges. In this method, a pre-smoothing step is implemented to enhance the accuracy of edge detection in noisy images, and fast numerical algorithms are proposed to solve the model through a binary labeling process. Note that all of the aforementioned models that improve the MS model are based on the intensity information of the given image.

However, using the image intensity information alone may lead to undesirable results when the image has low quality, missing intensities or overlaps with complicated background. As a remedy, prior knowledge of shapes can be taken into consideration in these situations [26–28]. Especially in real-world applications, the shapes of objects are known *a priori* in an image.

Recently, shape-based segmentation approaches have drawn considerable attention (see e.g., [29]), especially for medical images [30,31]. Tsai et al. proposed a parametric model to implicitly represent the segmentation curve [32], but the method can only control uni-modal shape densities. On the other hand, nonparametric methods have been proposed to handle multi-modal shape densities [33,34]. Furthermore, Gloger et al. [30] and [35] improved the existing technique for prior shape level set segmentation to delineate the liver in tissue-specific liver probability maps. These methods typically minimize an energy function containing regularization, data fidelity and shape terms. More recently, Guo et al. proposed the Automatic Prior Shape Selection (APSS) method for image segmentation based on the sparse representation of a dictionary containing prior shapes [36]:

$$\min_{u, v, s, d, h} \left\{ \frac{1}{2} \|u - I\|_2^2 + \frac{\eta}{2} \|v \cdot \nabla u\|_2^2 + \frac{\rho}{2} \|\nabla v\|_2^2 + \frac{\|v - 1\|_2^2}{8\rho} + \theta \|\nabla w\|_1 + \zeta \|s\|_1 + W(h) \right\}, \quad (3)$$

subject to $As + w = v(h)$.

Here v is the edge strength function of the image with range $[0, 1]$, w is discrepancy, A is a dictionary that contains all prior shapes, s contains the weights of prior shapes, h is the transformation between the detected edge and the approximated shape represented by the dictionary, $W(h)$ is a regularization term with respect to h , and $\eta, \rho, \theta, \zeta$ are positive tuning parameters. However, since the detected edges especially on the corners are sometimes thick (see Figs. 2(b) and 2(f)), one more post-processing step, i.e., thresholding the edge strength function v , is required to get the desired thin edges (see Figs. 2(c) and 2(g)).

Inspired by Shi et al. [25] and Guo et al. [36], we propose a new edge detection model whose energy functional has the advantages of accurate boundaries and prior shape information. This paper makes the following four major contributions. Firstly, we present a novel variational model that takes advantage of the prior shape information together with the intensity information for edge detection. The proposed model characterizes the relation between the prior shapes and the detected edges. In fact, this method allows the adaptive learning of shapes through the iterative algorithm. Secondly, we provide theoretical convergence guarantees of the proposed algorithm. Note that the standard ADMM convergence analysis is not applicable here since the objective function in (5) is not jointly convex with respect to all variables. Thirdly, our approach is robust which can deal with translation, uniform scaling and rotation of prior shapes in the given dictionary, and it also works for multi-shape images. Lastly, the proposed framework is able to generate the desired edges by directly solving a minimization problem without any post-processing steps.

The rest of the paper is organized as follows. In Section 2, we propose our new edge detection model using prior shapes. In Section 3, we describe in detail an efficient algorithm based on ADMM for solving the minimization problem (5). Section 4 provides convergence analysis of the proposed algorithm. Section 5 shows a variety of numerical experiments to demonstrate effectiveness and robustness of the proposed approach in edge detection. Concluding remarks and future work are given in Section 6.

2. The proposed model

Before presenting the edge detection model using prior shape that we proposed in this paper, we need to introduce the following notations. For the convenience, we vectorize an image by column-wise stacking.

Let $I \in \mathbb{R}^{\tilde{m}\tilde{n}}$ be the vectorization of the reference image, $u \in \mathbb{R}^{\tilde{m}\tilde{n}}$ a piecewise smooth approximation of I , and $\psi \in [0, 1]^{\tilde{m}\tilde{n}} := \Gamma$ an edge strength function, i.e., a convex relaxation of the binary edge indicator function. Assume that $A \in \mathbb{R}^{\tilde{m}\tilde{n} \times l}$ is a dictionary which consists of l prior shapes such that

$$As + w + e = \psi(h), \quad (4)$$

where w is the discrepancy between $\psi(h)$ and the approximated edge using the dictionary, h is the geometric transformation and e is additive noise. That is, the discrete noise-free edge strength function $\psi \in \mathbb{R}^{\tilde{m}\tilde{n}}$ can be decomposed into a sparse linear combination of prior shapes and a piecewise constant part after registration. If no geometric transformation is involved, then $\psi(h)$ is simply ψ .

Motivated by the models (2) and (3), we propose a new variational model using prior shapes for edge detection as follows:

$$\min_{u, w, s, \psi \in \Gamma} \left\{ \mu \|(1 - \psi) \odot |Du|\|_2^2 + \frac{\nu}{2} \|u - I\|_2^2 + \tau \|\psi\|_1 + \|D\psi\|_1 + \beta \|Dw\|_1 + \alpha \|s\|_1 + \frac{\gamma}{2} \|As + w - \psi\|_2^2 \right\}. \quad (5)$$

Here $D = [D_1^T, D_2^T]^T \in \mathbb{R}^{2\tilde{m}\tilde{n} \times \tilde{m}\tilde{n}}$ is the discrete gradient operator with the symmetric boundary conditions, $\odot$ is componentwise multiplication of two vectors.

The first term in the above model (5) can be further written as

$$\|(1 - \psi) \odot |Du|\|_2^2 = \|D_\psi D_1 u\|_2^2 + \|D_\psi D_2 u\|_2^2. \quad (6)$$

Here $D_\psi = E - \text{diag}(\psi)$, where E is the identity matrix and $\text{diag}(\cdot)$ is the diagonalization operator which turns a vector into a diagonal matrix. The model is robust to noise since that there are three total variation terms in the proposed model, where the first, fourth, fifth terms which allow us to smoothen homogeneous areas of the recovered image while preserving sharp edges. The last two terms in the proposed model (5) can be considered as shape terms, which essentially evaluate the shape difference among the prior shapes in the given dictionary. One can see that seeking the solution of (5) corresponds to the learning of edges that best fit the given prior shapes. In other words, the minimizer is achieved when the prior shapes have been fully learned through an iterative algorithm.

In this paper, we consider three basic types of geometric transformations: translation, uniform scaling and rotation. As rigid transformations, translation and rotation do not change lengths and angle measures. The uniform scaling transformation stretches or shrinks a given object and thereby changes lengths and angles. However, the shape of a geometric object will not be changed. That is, planar curves including lines, circles and ellipsoids are not changed under the aforementioned transformations. But the position and orientation of the object will be changed. It implies that these three types of transformations can be easily controlled. Further, more general affine transformations will be considered in our future work.

3. Proposed algorithm

We focus in this section on the numerical algorithm of a solution of problem (5). A lot of algorithms have been proposed in the literature for solving a similar problem, such as the Fast Total Variation deconvolution (FTVd) algorithm [37], fixed point algorithm [38,39], Algebraic Multigrid (AMG) algorithm [40], ADMM [41], dual algorithm [42], gradient projection [43,44], domain decomposition [45,46]. Many algorithms attempt to solve the edge detection problem in a generalized level set framework to obtain numerical results easily. With no intent of being exhaustive, interested readers are referred to [20,25,47,48] and the references therein for a comprehensive study. Since ADMM can solve decomposable minimization problems efficiently and even lead to parallelization for large-scale problems, we design an algorithm by applying ADMM to solve the proposed model (5). The proposed algorithm decomposes the original sophisticated problem into several simple subproblems, whose solutions can be easily obtained with relatively low computational cost.

Similar to the methods in [49,50], we decompose the original problem into several simpler subproblems and solve them in an alternating fashion. More precisely, by introducing the new variables $\mathbf{x} = (x_1, x_2)$, $\mathbf{y}$, $\mathbf{z}$, q , we can reformulate (5) as

$$\begin{aligned} \min_{\substack{u, w, s, \psi \\ x_1, x_2, \mathbf{y}, \mathbf{z}}} \left\{ \mu \|x_1\|_2^2 + \mu \|x_2\|_2^2 + \frac{\nu}{2} \|u - I\|_2^2 + \tau \|q\|_1 + \|\mathbf{y}\|_1 + \beta \|\mathbf{z}\|_1 + \alpha \|s\|_1 \right. \\ \left. + \frac{\gamma}{2} \|As + w - \psi\|_2^2 \right\}, \\ \text{subject to } D_\psi D_1 u = x_1, \quad D_\psi D_2 u = x_2, \quad D\psi = \mathbf{y}, \quad Dw = \mathbf{z}, \quad \psi = q. \end{aligned} \quad (7)$$

Then we define the augmented Lagrangian function

$$\begin{aligned} \mathcal{L} = & \mu \|x_1\|_2^2 + \mu \|x_2\|_2^2 + \frac{\nu}{2} \|u - I\|_2^2 + \tau \|q\|_1 + \|\mathbf{y}\|_1 + \beta \|\mathbf{z}\|_1 + \alpha \|s\|_1 \\ & + \frac{\gamma}{2} \|As + w - \psi\|_2^2 + \frac{\rho_1}{2} \sum_{m=1}^2 \|D_\psi D_m u - x_m + \tilde{x}_m\|_2^2 + \frac{\rho_2}{2} \|D\psi - \mathbf{y} + \tilde{\mathbf{y}}\|_2^2 \\ & + \frac{\rho_3}{2} \|Dw - \mathbf{z} + \tilde{\mathbf{z}}\|_2^2 + \frac{\rho_4}{2} \|\psi - q + \tilde{q}\|_2^2. \end{aligned} \quad (8)$$

There are eight variables to be determined in (8), which lead to the following eight subproblems with respect to u , ψ , q , s , w , x_m , $\mathbf{y}$, $\mathbf{z}$, respectively:

- u -subproblem: For the fixed ψ , x_m and $\tilde{x}_m$, we solve

$$\min_u \left\{ \mathfrak{R}(u) := \frac{\nu}{2} \|u - I\|_2^2 + \frac{\rho_1}{2} \sum_{m=1}^2 \|D_\psi D_m u - x_m + \tilde{x}_m\|_2^2 \right\}. \quad (9)$$

- ψ -subproblem: For the fixed u , q , $\tilde{q}$, s , w , x_m , $\tilde{x}_m$, $\mathbf{y}$ and $\tilde{\mathbf{y}}$, we solve

$$\min_\psi \left\{ S(\psi) := \frac{\rho_1}{2} \sum_{m=1}^2 \|D_\psi D_m u - x_m + \tilde{x}_m\|_2^2 + \frac{\rho_2}{2} \|D\psi - \mathbf{y} + \tilde{\mathbf{y}}\|_2^2 + \frac{\rho_4}{2} \|\psi - q + \tilde{q}\|_2^2 + \frac{\gamma}{2} \|As + w - \psi\|_2^2 \right\}. \quad (10)$$

- q -subproblem: For the fixed ψ and $\tilde{q}$, we solve

$$\min_q \tau \|q\|_1 + \frac{\rho_4}{2} \|\psi - q + \tilde{q}\|_2^2. \quad (11)$$

- s -subproblem: For the fixed w and ψ , we solve

$$\min_s \alpha \|s\|_1 + \frac{\gamma}{2} \|As + w - \psi\|_2^2. \quad (12)$$

- w -subproblem: For the fixed s , $\mathbf{z}$, $\tilde{\mathbf{z}}$ and ψ , we solve

$$\min_w \frac{\gamma}{2} \|As + w - \psi\|_2^2 + \frac{\rho_3}{2} \|Dw - \mathbf{z} + \tilde{\mathbf{z}}\|_2^2. \quad (13)$$

- x_m -subproblem: For the fixed ψ , u and $\tilde{x}_m$ with $m = 1, 2$, we solve

$$\min_{x_m} \mu \|x_m\|_2^2 + \frac{\rho_1}{2} \sum_{m=1}^2 \|D_\psi D_m u - x_m + \tilde{x}_m\|_2^2. \quad (14)$$

- $\mathbf{y}$ -subproblem: For the fixed ψ and $\tilde{\mathbf{y}}$, we solve

$$\min_{\mathbf{y}} \|\mathbf{y}\|_1 + \frac{\rho_2}{2} \|D\psi - \mathbf{y} + \tilde{\mathbf{y}}\|_2^2. \quad (15)$$

- $\mathbf{z}$ -subproblem: For the fixed w and $\tilde{\mathbf{z}}$, we solve

$$\min_{\mathbf{z}} \beta \|\mathbf{z}\|_1 + \frac{\rho_3}{2} \|Dw - \mathbf{z} + \tilde{\mathbf{z}}\|_2^2. \quad (16)$$

Despite the number of subproblems, each subproblem can be efficiently solved and some subproblems have the similar solutions. Specifically, the closed-form solutions of (11), (15), (16)-subproblems can be expressed using the shrinkage operator:

$$q = \text{sgn}(\psi + \tilde{q}) \odot \max \left\{ |\psi + \tilde{q}| - \frac{\tau}{\rho_4}, 0 \right\}. \quad (17)$$

$$\mathbf{y} = \text{sgn}(D\psi + \tilde{\mathbf{y}}) \odot \max \left\{ |D\psi + \tilde{\mathbf{y}}| - \frac{1}{\rho_2}, 0 \right\}. \quad (18)$$

$$\mathbf{z} = \text{sgn}(Dw + \tilde{\mathbf{z}}) \odot \max \left\{ |Dw + \tilde{\mathbf{z}}| - \frac{\beta}{\rho_3}, 0 \right\}. \quad (19)$$

Here $\text{sgn}(\cdot)$ is the signum function which returns the componentwise sign of the input vector. Once u^{n+1} , x_m^{n+1} , ψ^{n+1} , $\mathbf{y}^{n+1}$, w^{n+1} and $\mathbf{z}^{n+1}$ are available, the Lagrangian multipliers $\tilde{x}_m$, $\tilde{\mathbf{y}}$, $\tilde{\mathbf{z}}$ and $\tilde{q}$ can be updated via the following formulas:

$$\tilde{x}_m^{n+1} = \tilde{x}_m^n + (D_{\psi^{n+1}} D_m u^{n+1} - x_m^{n+1}), \quad m = 1, 2, \quad (20)$$

$$\tilde{\mathbf{y}}^{n+1} = \tilde{\mathbf{y}}^n + (D\psi^{n+1} - \mathbf{y}^{n+1}), \quad (21)$$

$$\tilde{\mathbf{z}}^{n+1} = \tilde{\mathbf{z}}^n + (Dw^{n+1} - \mathbf{z}^{n+1}), \quad (22)$$

$$\tilde{q}^{n+1} = \tilde{q}^n + (\psi^{n+1} - q^{n+1}). \quad (23)$$

The detailed process can be referred to [25,51,52]. Moreover, the s -subproblem (12) is a Lasso type of problem which can be directly solved [41,53]. Below we elaborate on how to solve these four subproblems about (9), (10), (13) and (14). Here we apply the fixed-point iteration method to solve the u -subproblem. ADMM is applied again to solve the ψ -subproblem and the w -subproblem. For the convenience of description, the image is written in matrix form in order to solve these subsequent subproblems. In what follows, we present the respective method in detail for solving the u -, ψ -, w - and x_m -subproblems. To impose boundary conditions, especially non-periodic ones, we resort to the PDE based methods by deriving an Euler–Lagrange equation in matrix form corresponding to the specific minimization problem.

3.1. Fixed-point iteration method for solving u

Due to the convexity of the objective function with respect to u , the problem (9) admits a unique minimizer. By considering an image as a BV function defined on a rectangular region $\Omega \subset \mathbb{R}^2$, we can get the following corresponding Euler–Lagrange equation with the Neumann boundary condition:

$$\begin{cases} -\rho_1 \operatorname{div}((1-\psi)^2 \nabla u) + \rho_1 \operatorname{div}((1-\psi)\hat{\mathbf{x}}) + v(u-I) = 0, & \text{in } \Omega, \\ \frac{\partial u}{\partial \mathbf{n}} \Big|_{\partial \Omega} = 0, & \text{on } \Omega, \end{cases} \quad (24)$$

where $\hat{\mathbf{x}} = (x_1 - \tilde{x}_1, x_2 - \tilde{x}_2) = (\hat{x}_1, \hat{x}_2)$ and $\mathbf{n}$ is the unit outer normal vector to the boundary $\partial \Omega$.

As a measure of edge strength, the function ψ takes small values in the homogeneous region, i.e., $1 - \psi \approx 1$. Hence, inspired by the ideas in [14,54], we use a fixed-point iteration method [39] based on the relaxation technique to solve the above elliptic equation. First we discretize the rectangular domain Ω by generating a uniform grid over Ω . By the finite difference scheme, we get the following discrete approximation of (24):

$$\begin{aligned} v u_{i,j} = & \rho_1 [(1-\psi)_{i,j+1}^2 (u_{i,j+1} - u_{i,j}) + (1-\psi)_{i,j-1}^2 (u_{i,j-1} - u_{i,j}) \\ & + (1-\psi)_{i-1,j}^2 (u_{i-1,j} - u_{i,j}) + (1-\psi)_{i+1,j}^2 (u_{i+1,j} - u_{i,j})] \\ & - \rho_1 [(1-\psi)_{i,j}(\hat{x}_1)_{i,j} - (1-\psi)_{i-1,j}(\hat{x}_1)_{i-1,j} \\ & + (1-\psi)_{i,j}(\hat{x}_2)_{i,j} - (1-\psi)_{i,j-1}(\hat{x}_2)_{i,j-1}] + v I_{i,j}, \end{aligned} \quad (25)$$

where $u_{i,j} \equiv u(i, j)$ is the approximate solution of (24) at the (i, j) th grid point. After applying the Gauss–Seidel method to the system of equations (25), we get

$$(\rho_1(C_E + C_W + C_N + C_S) + v)u_{i,j}^{k+1} = \rho_1 C' - \rho_1 \hat{F} + v I_{i,j}, \quad (26)$$

where

$$\begin{aligned} C_E &= (1-\psi^k)_{i,j+1}^2, & C_W &= (1-\psi^k)_{i,j-1}^2, \\ C_N &= (1-\psi^k)_{i-1,j}^2, & C_S &= (1-\psi^k)_{i+1,j}^2, \\ C' &= C_E u_{i,j+1}^k + C_W u_{i,j-1}^{k+1} + C_N u_{i-1,j}^{k+1} + C_S u_{i+1,j}^k, \\ \hat{F} &= [(1-\psi^k)_{i,j}(\hat{x}_1^k)_{i,j} - (1-\psi^k)_{i-1,j}(\hat{x}_1^k)_{i-1,j} \\ &+ (1-\psi^k)_{i,j}(\hat{x}_2^k)_{i,j} - (1-\psi^k)_{i,j-1}(\hat{x}_2^k)_{i,j-1}]. \end{aligned}$$

To speed up the iteration (26), we can use the relaxation technique. Let

$$u_{i,j}^{k+1} = u_{i,j}^k - \omega_1 r_{i,j}^{k+1}, \quad (27)$$

where $\omega_1 > 0$ is the relaxation factor, and $r_{i,j}^{k+1}$ denotes the residue obtained by subtracting the right-hand side from the left-hand side in (26), which approaches zero in the relaxed scheme. Combining with (27), we get the following updating scheme:

$$u_{i,j}^{k+1} = \frac{u_{i,j}^k + \omega_1(v I_{i,j} + \rho_1 C' - \rho_1 \hat{F})}{1 + \omega_1[\rho_1(C_E + C_W + C_N + C_S) + v]}. \quad (28)$$

3.2. ADMM algorithm for solving ψ

The minimization problem (10) is a typical variational model which uses the total variation as the regularization term and the L_2 fidelity term. Now we denote $\mathbf{y} = (y_1, y_2)$ and $\tilde{\mathbf{y}} = (\tilde{y}_1, \tilde{y}_2)$. Then we can get the following equivalent iterative

scheme

$$\begin{aligned}
 & (\psi^{n+1}, y_1^{n+1}, y_2^{n+1}) \\
 &= \operatorname{argmin}_{\psi, y_1, y_2} \frac{\rho_1}{2} \sum_{m=1}^2 \|D_\psi D_m u - x_m + \tilde{x}_m\|_2^2 + \frac{\rho_2}{2} \|D_1 \psi - y_1 + \tilde{y}_1^n\|_2^2 \\
 & \quad + \frac{\rho_2}{2} \|D_2 \psi - y_2 + \tilde{y}_2^n\|_2^2 + \frac{\rho_4}{2} \|\psi - q + \tilde{q}\|_2^2 + \frac{\gamma}{2} \|As + w - \psi\|_2^2.
 \end{aligned} \tag{29}$$

and

$$\tilde{y}_1^{n+1} = \tilde{y}_1^n + (D_1 \psi^{n+1} - y_1^{n+1}), \tag{30}$$

$$\tilde{y}_2^{n+1} = \tilde{y}_2^n + (D_2 \psi^{n+1} - y_2^{n+1}). \tag{31}$$

If we fix y_1 and y_2 , the discrete Euler–Lagrange equation of (29) for ψ can be written as

$$\begin{aligned}
 & \rho_1 \sum_{m=1}^2 (D_m u)^* (D_m u) \psi - \rho_1 \sum_{m=1}^2 (D_m u)^* (D_m u) + \rho_1 \sum_{m=1}^2 (D_m u)^* (x_m - \tilde{x}_m) \\
 & - \rho_2 \Delta \psi + \rho_4 (\psi - q + \tilde{q}) - \rho_2 D_1^* (y_1 - \tilde{y}_1^n) - \rho_2 D_2^* (y_2 - \tilde{y}_2^n) - \gamma (As + w - \psi) = 0.
 \end{aligned} \tag{32}$$

Following the same idea as the u -subproblem, we can apply the relaxation technique and get the following updating scheme for ψ :

$$\psi_{i,j}^{k+1} = \frac{\psi_{i,j}^k + \omega_2 [F_{i,j}^k + \rho_2 (\psi_{i+1,j}^k + \psi_{i-1,j}^{k+1} + \psi_{i,j-1}^{k+1} + \psi_{i,j+1}^k)]}{1 + \omega_2 \left[\rho_1 \left(\sum_{m=1}^2 (D_m u)^* (D_m u) \right)_{i,j}^{k+1} + 4\rho_2 + \rho_4 + \gamma \right]}, \tag{33}$$

where $\omega_2 > 0$ is the relaxation factor and

$$\begin{aligned}
 F &:= \rho_1 \sum_{m=1}^2 (D_m u)^* (D_m u) + \rho_4 (q - \tilde{q}) + \rho_2 D_1^* (y_1 - \tilde{y}_1^n) + \rho_2 D_2^* (y_2 - \tilde{y}_2^n) \\
 & + \gamma (As + w) - \rho_1 \sum_{m=1}^2 (D_m u)^* (x_m - \tilde{x}_m).
 \end{aligned} \tag{34}$$

Similar to [48], to further enforce $\psi \in [0, 1]$, we define the truncation function

$$\operatorname{Tr}(\psi) = \begin{cases} 1, & \psi > 1; \\ \psi, & 0 \leq \psi \leq 1; \\ 0, & \psi < 0, \end{cases} \tag{35}$$

which satisfies that $\psi \in [0, 1]$.

3.3. ADMM algorithm for solving w

Similar to the ψ -subproblem, we can apply ADMM to get the solution w . We let $\mathbf{z} = (z_1, z_2)$ and $\tilde{\mathbf{z}} = (\tilde{z}_1, \tilde{z}_2)$ and use the same parameters ρ_3, γ as before. Here we skip the derivation but show the Euler–Lagrange equation for w as follows:

$$- \rho_3 \Delta w - \rho_3 D_1^* (z_1 - \tilde{z}_1^n) - \rho_3 D_2^* (z_2 - \tilde{z}_2^n) + \gamma (As + w - \psi) = 0, \tag{36}$$

Thus, the solution w of (36) is given by

$$w_{i,j}^{k+1} = \frac{\tilde{F}_{i,j}^k + \rho_3 (w_{i+1,j}^k + w_{i-1,j}^{k+1} + w_{i,j-1}^{k+1} + w_{i,j+1}^k)}{4\rho_3 + \gamma}, \tag{37}$$

where

$$\tilde{F} := \rho_3 D_1^* (z_1 - \tilde{z}_1^n) + \rho_3 D_2^* (z_2 - \tilde{z}_2^n) - \gamma (As - \psi). \tag{38}$$

Notice that there is no need to relax the w -subproblem, since the coefficient matrix of the linear system is positive definite.

3.4. Fixed-point iteration method for solving x_m

Following the same idea as the u -subproblem, we can get the following Euler–Lagrange equation with respect to x_1 and x_2 :

$$2\mu x_m - \rho_1 [(1 - \psi)D_m u - x_m + \tilde{x}_m] = 0, \quad m = 1, 2. \quad (39)$$

Then we have the following updating scheme for x_m with $m = 1, 2$:

$$(x_m)_{i,j}^{k+1} = \frac{\rho_1 [(1 - \psi_{i,j}^{k+1})(D_m u)_{i,j}^{k+1} + (\tilde{x}_m)_{i,j}^k]}{2\mu + \rho_1}. \quad (40)$$

In summary, the proposed algorithm to solve the minimization problem (8) is described in the Algorithm 1.

Algorithm 1 Image edge detection with automatic prior shape selection

Inputs: reference image u^0 , parameters $\mu, \nu, \tau, \beta, \alpha, \gamma, \rho_1, \rho_2, \rho_3$, and ρ_4 .

Initialization: set $\psi^0 = 0, s^0 = 0, w^0 = 0, y_1^0 = y_2^0 = \tilde{y}_1^0 = \tilde{y}_2^0 = 0, \tilde{x}_m = 0, x_m = 0, \tilde{q}^0 = 0$,

For $n = 0, 1, \dots$,

- (a) Compute u^{n+1} by solving (28);
- (b) Compute ψ^{n+1} by solving (33), then set $\psi^{n+1} = \text{Tr}(\psi^{n+1})$;
- (c) Compute w^{n+1} by solving (37);
- (d) Compute s^{n+1} by solving the Lasso problem;
- (e) Compute $\mathbf{y}^{n+1}$ by the shrinkage formula (18);
- (f) Update $\tilde{\mathbf{y}}^{n+1}$ by (21);
- (g) Compute $\mathbf{z}^{n+1}$ by the shrinkage formula (19);
- (h) Update $\tilde{\mathbf{z}}^{n+1}$ by (22);
- (i) Compute q^{n+1} by the shrinkage formula (17);
- (j) Update $\tilde{q}^{n+1}$ by (23);
- (k) Compute x_m^{n+1} by solving (40);
- (l) Update $\tilde{x}_m^{n+1}$ by (20).

Exit when the stopping criteria are met.

4. Convergence analysis

In this section, we discuss the convergence of the proposed Algorithm 1. Since the entire algorithm has “inner” and “outer” iterations, we first prove the convergence of the generated sequence to the solution of each subproblem. Then we show the convergence of the proposed algorithm to the minimizer of the problem (5). Note that the convergence rate of the Lasso problem has been discussed in [55]. Therefore, we can focus on the u -subproblem, ψ -subproblem and w -subproblem and show the convergence of each iterative scheme. The detailed proofs can be found in Appendix A.

Theorem 1. The sequence $\{u^k\}_{k \geq 0}$ (resp. $\{\psi^k\}_{k \geq 0}$ or $\{w^k\}_{k \geq 0}$) generated by the inner iterative scheme (28) (resp. (33) or (37)) converges to the solution u^* (resp. ψ^* or w^*) of the problem (9) (resp. (10) or (13)).

Following the argument for the convergence analysis in [49,50], we can show the convergence of the full ADMM-based algorithm. Notice that we can derive the first order optimality condition of (29) for the variables $u, \psi, \mathbf{y}$, respectively:

$$\begin{cases} \rho_1 D^*((1 - \psi^*)^2 D u^{n+1}) + \tilde{u}^{n+1} = 0, \\ -\rho_2 \Delta \psi^{n+1} + \rho_1 \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) (\psi^{n+1} - 1) + \tilde{\psi}^{n+1} - \rho_2 D^*(\mathbf{y}^n - \tilde{\mathbf{y}}^n) = 0, \\ \mathbf{p}^{n+1} + \rho_2 (\mathbf{y}^{n+1} - D \psi^{n+1} - \tilde{\mathbf{y}}^n) = 0, \\ \tilde{\mathbf{y}}^{n+1} = \tilde{\mathbf{y}}^n + (D \psi^{n+1} - \mathbf{y}^{n+1}). \end{cases} \quad (41)$$

where

$$\begin{aligned} \tilde{u}^{n+1} &= -\rho_1 (1 - \psi^*) \sum_{m=1}^2 D_m^* (x_m^* - \tilde{x}_m^*) + \partial \left(\frac{\nu}{2} \|u^{n+1} - I\|_2^2 \right); \\ \tilde{\psi}^{n+1} &= \rho_1 \sum_{m=1}^2 (D_m u^*)^* (x_m^* - \tilde{x}_m^*) + \partial \left(\frac{\rho_4}{2} \|\psi^{n+1} - q^* + \tilde{q}^*\|_2^2 + \frac{\gamma}{2} \|A s^* + \omega^* - \psi^{n+1}\|_2^2 \right); \\ \mathbf{p}^{n+1} &\in \partial |\mathbf{y}^{n+1}|. \end{aligned}$$

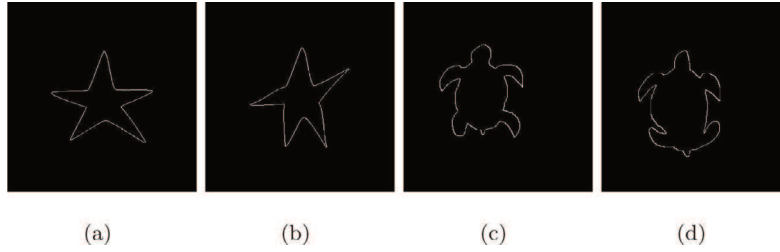


Fig. 1. The sample shapes in the dictionary A.

Theorem 2. Let (u^*, ψ^*) be the pair of minimizers of the subproblems (9)–(10). Given $\omega_1, \omega_2 > 0$ in (28) and (33), we have

$$\lim_{n \rightarrow +\infty} S(\psi^n) = S(\psi^*), \quad \lim_{n \rightarrow +\infty} \Re(u^n) = \Re(u^*), \quad (42)$$

where S and $\Re$ are the objective functions of the u - and ψ -subproblems, respectively. Moreover, if the pair of minimizers (u^*, ψ^*) is unique, we get

$$\lim_{n \rightarrow \infty} \|\psi^n - \psi^*\| = 0, \quad \lim_{n \rightarrow \infty} \|u^n - u^*\| = 0. \quad (43)$$

We provide the detailed proof in [Appendix B](#).

5. Numerical experiments

In this section, we provide a variety of numerical results to show the performance of the proposed model and algorithm. All experiments are implemented using Matlab (R2016a) on a laptop with Intel(R) Core(TM) i5-8250U @1.60 GHz, 8.0 GB RAM, Windows 10.

Edge detection can be considered as a process of learning fitting shapes from the prior shapes dictionary, which also corresponds to the iterative process of solving the minimization problem (5). The images containing prior shapes in the given dictionary are called the *training images*. Then the images that we want to detect edges are called the *test images*. A test image usually contains one or multiple test shapes. For example, [Fig. 2\(a\)](#) has one test shape and [Fig. 4\(h\)](#) has three test shapes. Every training image contains one prior shape (see [Fig. 1](#)). In order to describe the difference between the test image (resp. test shape) and the training image (resp. prior shape), here we consider three basic types of geometric transformations: translation, scaling and rotation. Especially, the scaling cases include two types. For the sake of discussion simplicity, we consider the following four cases for various test and training images:

S1. The size of the test image is the same as that of the training images. For example, the size of [Fig. 1\(a\)](#) is the same as that of [Fig. 2\(a\)](#).

S2. The size of the test shape in the test image is the same as that of the prior shapes in the training images. For example, the size of the star in [Fig. 2\(a\)](#) is the same as that of the star in [Fig. 1\(a\)](#).

T. The center of the shape in the test image lies in the same position as that of the prior shape in the training images. For example, the center of the star in [Fig. 2\(a\)](#) matches that of the star in [Fig. 1\(a\)](#).

R. The shape does not rotate an angle comparing with the prior shape in the training images (see the star in [Fig. 4\(a\)](#) and the star in [Fig. 1\(a\)](#)).

In the following examples, we set the stopping criteria by using the iteration numbers I_{in} , I_{out} and the relative error

$$\frac{\|u^{n+1} - u^n\|_1}{\|u^{n+1}\|_1} \leq \eta, \quad (44)$$

where $\eta > 0$, I_{in} is the number of inner iterations, I_{out} is the number of outer iterations. The iteration process is terminated when the prior shapes specified have been learned well. To show the robustness of our algorithm, we set $\eta = 10^{-4}$, $\mu = 1$, $\nu = 10^{-2}$, $\tau = 10^{-1}$, $\beta = 10^2$, $\alpha = 5 \times 10^{-5}$, $\gamma = 10^2$, $\omega_1 = 5 \times 10^{-4}$, $\omega_2 = 1$, $\rho_1 = 10^2$, $\rho_2 = 1$, $\rho_3 = 10^2$, $\rho_4 = 10^2$, $I_{in} = 10$, $I_{out} = 11$ in the all experiments. Since there are only few shapes in a test image, the dictionary A can be reduced in this experiment as long as the edge of the target object is detected. Our dictionary consists of 26 training images with different types of shapes, such as star and tortoise. All of the training images have the same size (356×320). Some of them are shown in [Fig. 1](#).

5.1. Experiment 1

To begin with, we consider the simplest case when the test image and the training images satisfy **S1**, **S2**, **T**, **R**.

As mentioned above, the proposed model (5) is closely related to the APSS [36]. So we first test two images from [36] to compare the results by our method and APSS. One image has a star partially occluded by the background rectangles,

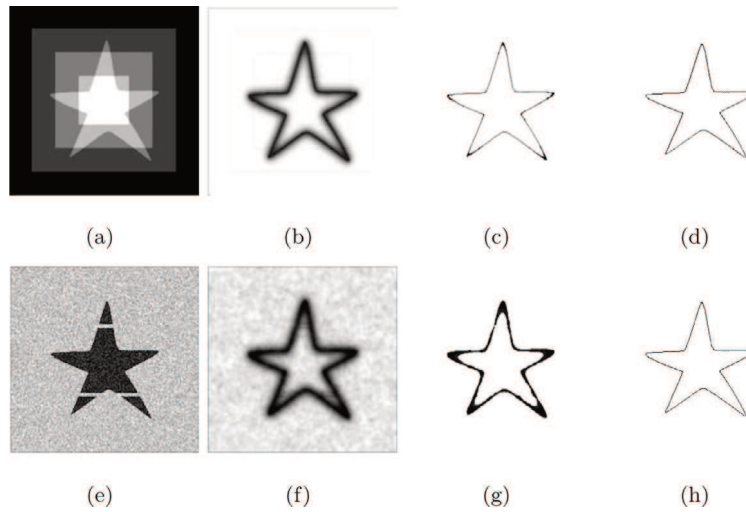


Fig. 2. Column 1: the input images; Column 2: the output edges strength function v by APSS (3); Column 3: the extracted edges being post-processed with APSS (3); Column 4: our detect edges $1 - \psi$ for the convenience of performance comparison.

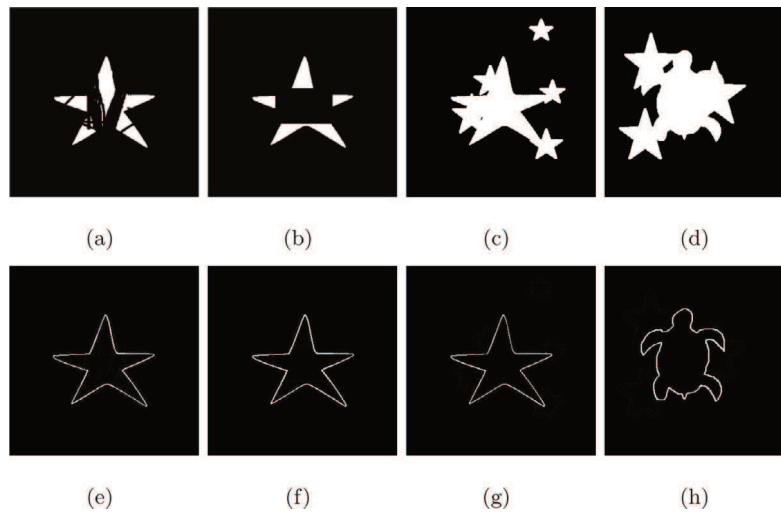


Fig. 3. Row 1: the contaminated images. Row 2: the detected edges ψ .

and the other image has a star with missing parts and contaminated by the Gaussian noise with zero mean and standard deviation $\delta = 0.8$.

In Fig. 2, one can see that our method and APSS can supplement the insufficiency of input data by learning a shape from the dictionary. However, our method yields results different from APSS in several aspects. Firstly, our result has clearer edges especially for the corners. Secondly, the edges of APSS need to be post-processed by hard thresholding but the proposed method can directly get thin edges without post-processing. To compare with APSS, we also visualize the detected edges by setting the image intensities as $1 - \psi$ in Fig. 2. Note that ψ is one on edges and zero otherwise in (5).

5.2. Experiment 2

In the second experiment, the test images and the training images still satisfy **S1**, **S2**, **T**, but fail to meet **R**. To show the robustness of our model, the test images are created by adding different kinds of degradations to the original image (see Fig. 3). Fig. 3(a) is a star degraded by many lines. The center of star in Fig. 3(b) is partly covered by a black area. Fig. 3(c) is surrounded and overlapped by some little stars which are not in the given dictionary. Fig. 3(d) is a tortoise, which is contaminated by some stars.

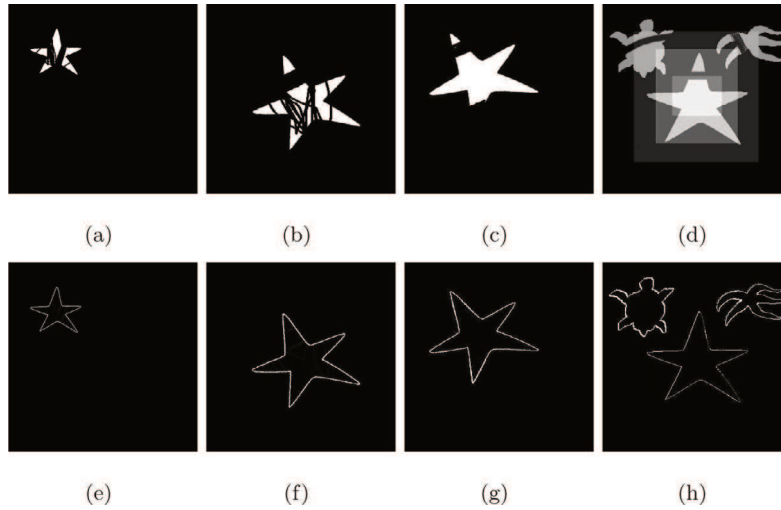


Fig. 4. Row 1: the contaminated images. Row 2: the detected edges ψ .

In Fig. 3(e), we can clearly find the missing part of the image in Fig. 3(a). The contaminated part (see Fig. 3(f)) is visible. For the images with complicated background (see Figs. 3(c) and 3(d)), our model can also extract the desired edges (see Figs. 3(g) and 3(h)).

5.3. Experiment 3

In practice, a test image may not have the same size as the training images in the dictionary. Therefore, we consider the test image and the training images do not meet **S2-T** (see Figs. 4(a), 4(c)). In Fig. 4(a), the test image is of 712×640 which is larger than the training images in the given dictionary, where each training image is of 356×320 . Fig. 4(b) is a star with rotation and contaminated by many lines. In Fig. 4(c), one angle is missing and the star is translated and rotated. Then we also test a multi-shape image with a complicated background, where the two different stars and one tortoise are included in the dictionary (see Fig. 4(d)). It implies that multi-shape image edge detection can be done by solely solving a minimization problem if the detected shapes are stored in the given dictionary in advance.

Fig. 4 shows that our model can get clear and complete edges from the training images with translation, scaling and rotation. Moreover, the edge of each shape can be detected with high accuracy in a multi-shape image as illustrated in Fig. 4(d).

5.4. Experiment 4

Fig. 5 shows the results of the four test images degraded by different noises. Figs. 5(a) and 5(b) are both contaminated by the Gaussian noise with zero mean and standard deviation $\delta = 0.1$. In Fig. 5(a), the test image is translated and contains a large region of missing intensities. Fig. 5(b) is contaminated by many lines without translation. Fig. 5(c) is degraded by the mixed noises which combines the Salt & pepper noise with density 0.3 and the Gaussian noise with zero mean and standard deviation $\delta = 0.3$. In Fig. 5(d), the test image containing three prior shapes is contaminated by the Salt & pepper noise with density 0.02 and the Gaussian noise with zero mean and standard deviation $\delta = 0.01$.

From Fig. 5, we can clearly observe that the edges are detected accurately with all noise suppressed. It shows that our model can deal with different types of noise, even excessive or mixed noises (see Fig. 5(c)). Since there is serious noise, some parameters are modified as $\mu = 90$, $\nu = 10^{-3}$, $\tau = 0.19$, $\alpha = 10^{-6}$, $\gamma = 80$, $\omega_1 = 6 \times 10^{-4}$, $\beta = 10^4$, $\rho_2 = 10^{-2}$ and $\rho_3 = 10^3$, and the others are the same as above.

5.5. Experiment 5

Fig. 7 shows the edge detection results of gray images with more gray levels. Fig. 7(a) is the original 'mushroom' image of size 356×320 that is downloaded from the Berkeley image segmentation dataset (see the website: http://imageprocessingplace.com/root_files_V3/image_databases.htm). The intensities of the 'mushroom' image range from 0 to 255. The top left of the 'mushroom' images in Figs. 7(c) and 7(d) are partly covered by a gray area. Figs. 7(b) and 7(d) are polluted by the mixed noises which are Salt & Pepper noise with density 0.02 and Gaussian noise with zero mean and standard deviation $\delta = 0.08$. From the detected edges (see the second row in Fig. 7), one can clearly observe that all noises have been removed and the edges are detected significantly and completely.

To evaluate the performance of the proposed algorithm, we also show the standard edge images from the dataset in Fig. 6. We can see that more detailed edges are detected.

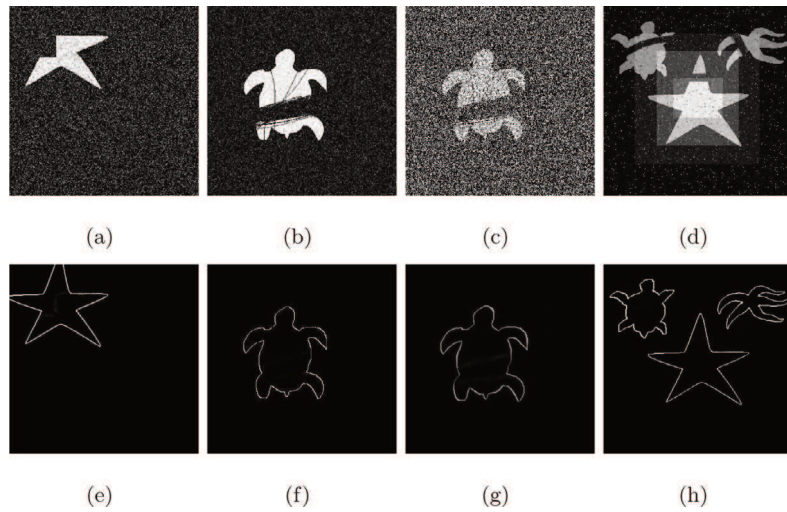


Fig. 5. Row 1: the contaminated images. Row 2: the detected edges ψ .

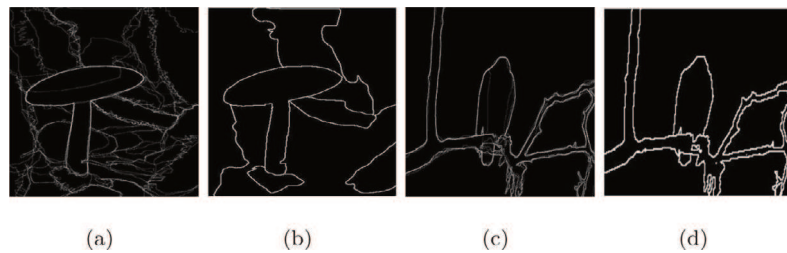


Fig. 6. The detected edges ψ in Berkeley image segmentation dataset.

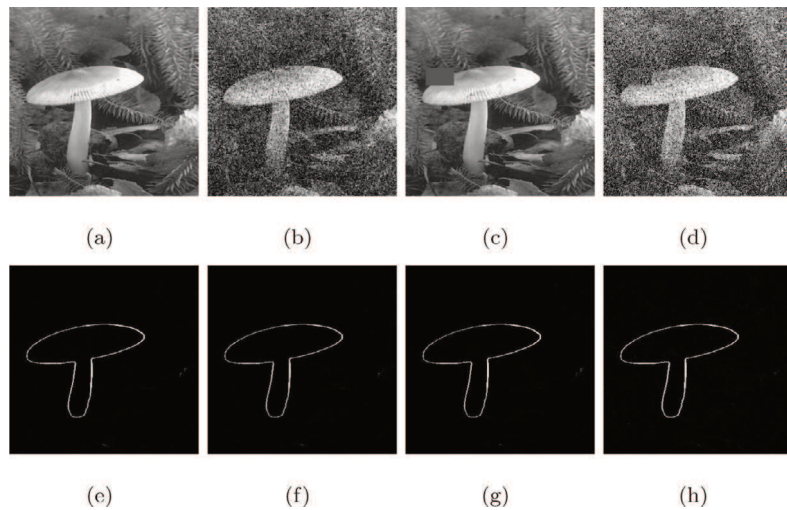


Fig. 7. Row 1: (a) the original image; (b) noisy image; (c) partly covered image; (d) partly covered image polluted by mixed noises. Row 2: the detected edges ψ .

5.6. Experiment 6

In the sixth experiment, the test images are the binary image (Fig. 8(a)) and the real 'bird' image (Fig. 8(e)) of size 356×320 that is also downloaded from the Berkeley image segmentation dataset. The intensities of the 'bird' image range from 0 to 255. Figs. 8(b) and 8(f) are the detected edges by our model when the dictionary contains the prior shape

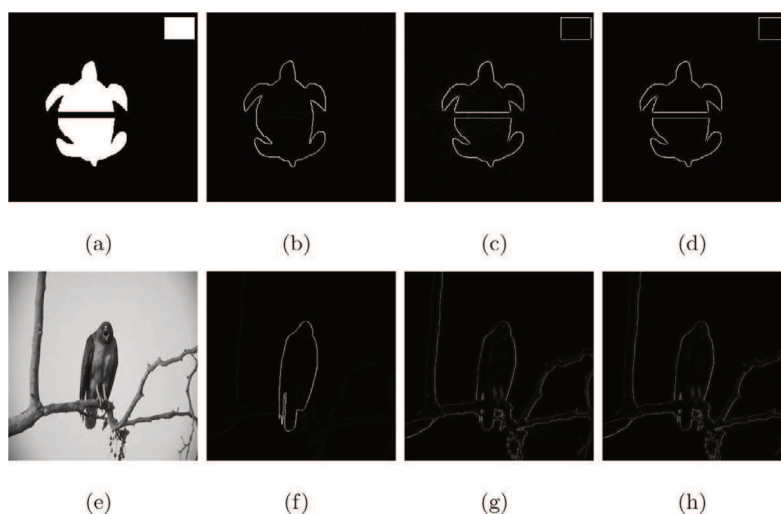


Fig. 8. Column 1: the original images; Column 2: detected edges by the proposed method with prior shape of the target in the library; Column 3: detected edges by the proposed method without prior shape of the target in the library; Column 4: detected edges by the proposed method without any prior in the library.

Table 1

Test times for different images.

Image	Fig. 3(b)	Fig. 4(c)	Fig. 5(h)	Fig. 6(a)	Fig. 6(b)	Fig. 6(c)	Fig. 6(d)	Fig. 7(a)	Fig. 7(h)
Time	20.65 s	19.15 s	13.89 s	14.30 s	20.76 s	20.49 s	18.98 s	16.27 s	16.30 s

of the target image. Figs. 8(c) and 8(g) are the detected edges when the dictionary does not contain any prior shape, that means the dictionary is empty. Figs. 8(d) and 8(h) show the detected edges when the dictionary does not contain the prior shape of the target image. It means that there is no shape in the dictionary to match the image, then s should be 0. Our algorithm is still able to detect all of the edges of the original image, not limited to the edge of the target (see the last columns of Fig. 8). Shapes that are not included in the dictionary could be removed as noise.

For the above six experiments, we compare computational times in Table 1. On the one hand, one can observe that it takes a little longer when we dealt with large coverage areas, rotations and mixed noises such as shown in Figs. 3(f), 4(g), 7(f) and 7(h). On the other hand, the whole experimental process is done using Matlab(R2016a) on a laptop with Intel(R) Core(TM) i5-8250U @1.60 GHz, 8.0 GB RAM, Windows 10. Due to the separable subproblems of the proposed algorithm, the computational time can be significantly reduced by using the parallel computing capability of GPUs, which is part of our future work.

Finally, we show that our approach is robust to the selection of parameters and tested a large amount of parameters. A basic selection guide for these parameters can be found in [25]. Larger μ or smaller ν , i.e., more weights on the regularization term, results in a smoother restored image and fewer edges in ψ , so we keep the optimal value of μ/ν in all experiments. The parameter τ is set to be less than 0.5 especially for noisy images. Larger β produces smoother restored image, so we typically set β to be larger than 10. If β is too small, then some edges may not be detected well such as the missing left-upper angle in Fig. 5(a). The smaller α implies sharper edges, so α is set to be less than 5×10^{-4} . The relaxation parameters ω_1, ω_2 usually are set to be smaller than 1. To sum up, various experiments have shown that the proposed approach is robust and can effectively generate edges with satisfactory visual qualities.

6. Conclusions

In this paper, we propose a novel variational model to detect edges in an image based on the prior shapes that are stored in a shape dictionary. Due to the complexity of the proposed model, we derive an efficient numerical algorithm based on ADMM. Each subproblem can either be converted to solving a Euler–Lagrange equation or be solved by the shrinkage operator. The detailed convergence analysis of the proposed algorithm is provided. A variety of numerical experiments show that the proposed method is able to extract edges from a given image with the guidance of prior shapes regardless of noise types and geometric transformations. It also performs well for the edge detection of an image with multiple objects. Furthermore, this framework can be extended for object recognition and various related real-world applications in the future.

Acknowledgments

The research is partially supported by NSFC, PR China (No. 11271126). The research of Jing Qin is supported by the National Science Foundation, USA grant DMS-1941197.

Appendix A. Proof of Theorem 1

The proof is motivated by the idea in [50]. Here only the convergence analysis for the u -subproblem is provided, as the other two subproblems can be discussed similarly. We begin the discussion by rewriting the iterative scheme (28) as

$$\begin{aligned} & [1 + \omega_1 (\rho_1(C_E + C_W + C_N + C_S) + \nu)] u_{i,j}^{k+1} - \rho_1 \omega_1 C_W u_{i,j-1}^{k+1} - \rho_1 \omega_1 C_N u_{i-1,j}^{k+1} \\ & = u_{i,j}^k + \rho_1 \omega_1 C_S u_{i+1,j}^k + \rho_1 \omega_1 C_E u_{i,j+1}^k + (\rho_1 \sum_{m=1}^2 D_\psi D_m^* (x_m - \tilde{x}_m))_{i,j}^k + \nu \omega_1 I_{i,j}, \end{aligned} \quad (45)$$

where

$$\begin{aligned} C_E &= (1 - \psi^k)_{i,j+1}^2, & C_W &= (1 - \psi^k)_{i,j-1}^2, \\ C_N &= (1 - \psi^k)_{i-1,j}^2, & C_S &= (1 - \psi^k)_{i+1,j}^2. \end{aligned}$$

Thus it can be rewritten in a matrix form

$$(L + \dot{D} + \dot{U})U^{k+1} = (E + \dot{D} + \dot{U})U^k + G + \hat{I}. \quad (46)$$

Here E is the identity matrix, U^k is obtained by stacking entries of u^k column-wise. Similarly, the vectors $G, \hat{I}$ are also obtained by stacking the corresponding matrix entries. The matrices $\dot{U}, \dot{U}, \dot{D}$ and $\dot{D}$ are lower or upper triangular matrices with the respective nonzero off-diagonal entries [25]. The matrix L is a diagonal matrix with diagonal entries

$$L_{i,j} = 1 + \omega_1 \rho_1 (C_E + C_W + C_N + C_S) + \omega_1 \nu. \quad (47)$$

Suppose that $\hat{\lambda}$ is an eigenvalue of the iteration matrix

$$A = (L + \dot{D} + \dot{U})^{-1} (E + \dot{D} + \dot{U}) = (a_{i,j}). \quad (48)$$

According to the Hadamard theorem, there exists one integer k such that

$$|a_{kk} \hat{\lambda}| \leq |\hat{\lambda}| \sum_{j=1}^{k-1} |a_{kj}| + \sum_{j=k+1}^n |a_{kj}|. \quad (49)$$

Thus, we have $\hat{\lambda} < 1$.

That implies $\lim_{k \rightarrow \infty} u^k = u^*$. The convergence discussions about the sequences $\{\psi^k\}$ and $\{w^k\}$ follow the same argument as above.

Appendix B. Proof of Theorem 2

Let ψ^* be the exact solution of the problem (10). Considering the first order optimality condition, ψ^* satisfies

$$\rho_1 \sum_{m=1}^2 ((D_m u^*)^* (D_m u^*)) (\psi^* - 1) + \partial \left(\frac{\rho_2}{2} \|D\psi^* - \mathbf{y} + \tilde{\mathbf{y}}\|_2^2 \right) + \tilde{\psi}^* = 0, \quad (50)$$

for the fixed u^* and

$$\tilde{\psi}^* = \rho_1 \sum_{m=1}^2 (D_m u^*)^* (x_m^* - \tilde{x}_m^*) + \partial \left(\frac{\rho_4}{2} \|\psi^* - q + \tilde{q}\|_2^2 + \frac{\gamma}{2} \|As^* + \omega^* - \psi^*\|_2^2 \right). \quad (51)$$

Similarly, let u^* be an arbitrary exact solution of (9). By the first order optimality condition, u^* satisfies

$$\rho_1 D^* ((\psi^* - 1)^2 Du) + \tilde{u}^* = 0, \quad (52)$$

with the fixed ψ^* and

$$\tilde{u}^* = -\rho_1 (1 - \psi^*) \sum_{i=1}^2 D_m^* (x_m^* - \tilde{x}_m^*) + \partial \left(\frac{\nu}{2} \|u^* - I\|_2^2 \right). \quad (53)$$

Therefore, we have that $(u^*, \psi^*, \mathbf{y}^*, \mathbf{p}^*)$ must satisfy

$$\begin{cases} \rho_1 D^* ((1 - \psi^*)^2 Du^*) + \tilde{u}^* = 0, \\ -\rho_2 \Delta \psi^* + \rho_1 \left(\sum_{m=1}^2 (\psi^* - 1)(D_m u^*)^* (D_m u^*) \right) + \tilde{\psi}^* - \rho_2 D^* (\mathbf{y}^* - \tilde{\mathbf{y}}^*) = 0, \\ \mathbf{p}^* + \rho_2 (\mathbf{y}^* - D\psi^* - \tilde{\mathbf{y}}^*) = 0, \\ \tilde{\mathbf{y}}^* = \tilde{\mathbf{y}}^* + (D\psi^* - \mathbf{y}^*). \end{cases} \quad (54)$$

The above Eqs. (54) show that $(u^*, \psi^*, \mathbf{y}^*, \mathbf{p}^*)$ is a fixed point of (41). Denote the errors by

$$\begin{aligned} u_e^n &= u^n - u^*, \quad \psi_e^n = \psi^n - \psi^*, \\ \tilde{u}_e^n &= \tilde{u}^n - \tilde{u}^*, \quad \tilde{\psi}_e^n = \tilde{\psi}^n - \tilde{\psi}^*, \\ \mathbf{p}_e^n &= \mathbf{p}^n - \mathbf{p}^*, \quad \tilde{\mathbf{y}}_e^n = \tilde{\mathbf{y}}^n - \tilde{\mathbf{y}}^*, \\ \mathbf{y}_e^n &= \mathbf{y}^n - \mathbf{y}^*. \end{aligned}$$

Subtracting the second equation of (41) from the second equation of (54), we obtain

$$-\rho_2 \Delta \psi_e^{n+1} + \rho_1 \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) ((\psi^{n+1} - 1) - (\psi^* - 1)) + \tilde{\psi}_e^{n+1} - \rho_2 D^* (\mathbf{y}_e^n - \tilde{\mathbf{y}}_e^n) = 0. \quad (55)$$

By taking the inner product of both sides of the above equation with ψ_e^{n+1} , we get

$$\begin{aligned} \rho_2 \|D\psi_e^{n+1}\|^2 + \rho_1 \|\psi_e^{n+1}\|^2 \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) + \langle \tilde{\psi}_e^{n+1}, \psi_e^{n+1} \rangle \\ - \langle \rho_2 D^* (\mathbf{y}_e^n - \tilde{\mathbf{y}}_e^n), \psi_e^{n+1} \rangle = 0. \end{aligned} \quad (56)$$

Applying the same technique to the third equation of (41) and the third equation of (54), we obtain

$$\langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle + \rho_2 \|\mathbf{y}_e^{n+1}\|^2 - \rho_2 \langle \mathbf{y}_e^{n+1}, D\psi_e^{n+1} + \tilde{\mathbf{y}}_e^n \rangle = 0. \quad (57)$$

Summing up (56) and (57), we get

$$\begin{aligned} \rho_2 \|D\psi_e^{n+1}\|^2 + \rho_1 \|\psi_e^{n+1}\|^2 \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) + \langle \tilde{\psi}_e^{n+1}, \psi_e^{n+1} \rangle + \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle \\ + \rho_2 \|\mathbf{y}_e^{n+1}\|^2 - \rho_2 \langle D\psi_e^{n+1}, \mathbf{y}_e^{n+1} + \mathbf{y}_e^n \rangle + \rho_2 \langle D\psi_e^{n+1} - \mathbf{y}_e^{n+1}, \tilde{\mathbf{y}}_e^n \rangle = 0. \end{aligned} \quad (58)$$

Furthermore, by subtracting the fourth equation of (41) from the fourth equation of (54), we obtain

$$\tilde{\mathbf{y}}_e^{n+1} = \tilde{\mathbf{y}}_e^n + D\psi_e^{n+1} - \mathbf{y}_e^{n+1}, \quad (59)$$

which leads to

$$\|\tilde{\mathbf{y}}_e^{n+1}\|^2 = \|\tilde{\mathbf{y}}_e^n\|^2 + \|D\psi_e^{n+1} - \mathbf{y}_e^{n+1}\|^2 + 2\langle \tilde{\mathbf{y}}_e^n, D\psi_e^{n+1} - \mathbf{y}_e^{n+1} \rangle. \quad (60)$$

That is,

$$\langle \tilde{\mathbf{y}}_e^n, D\psi_e^{n+1} - \mathbf{y}_e^{n+1} \rangle = \frac{1}{2} (\|\tilde{\mathbf{y}}_e^{n+1}\|^2 - \|\tilde{\mathbf{y}}_e^n\|^2) - \frac{1}{2} \|D\psi_e^{n+1} - \mathbf{y}_e^{n+1}\|^2. \quad (61)$$

Substituting (61) into (58), we have

$$\begin{aligned} \frac{\rho_2}{2} (\|\tilde{\mathbf{y}}_e^n\|^2 - \|\tilde{\mathbf{y}}_e^{n+1}\|^2) \\ = \rho_2 \|D\psi_e^{n+1}\|^2 + \rho_1 \|\psi_e^{n+1}\|^2 \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) + \langle \tilde{\psi}_e^{n+1}, \psi_e^{n+1} \rangle \\ + \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle + \rho_2 \|\mathbf{y}_e^{n+1}\|^2 - \langle \rho_2 D\psi_e^{n+1}, \mathbf{y}_e^{n+1} + \mathbf{y}_e^n \rangle - \frac{\rho_2}{2} \|D\psi_e^{n+1} - \mathbf{y}_e^{n+1}\|^2. \end{aligned} \quad (62)$$

Then, by summing up the above inequalities from 0 to N , we obtain

$$\begin{aligned} \frac{\rho_2}{2} (\|\tilde{\mathbf{y}}_e^0\|^2 - \|\tilde{\mathbf{y}}_e^{N+1}\|^2) \\ = \rho_1 \sum_{n=0}^N \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) \|\psi_e^{n+1}\|^2 + \sum_{n=0}^N (\langle \tilde{\psi}_e^{n+1}, \psi_e^{n+1} \rangle \end{aligned}$$

$$+ \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle + \frac{\rho_2}{2} \left(\sum_{n=0}^N \|D\psi_e^{n+1} - \mathbf{y}_e^n\|^2 + \|\mathbf{y}_e^{N+1}\|^2 \right) - \frac{\rho_2}{2} \|\mathbf{y}_e^0\|^2. \quad (63)$$

Noting that all terms involved in (63) are nonnegative, and the facts $0 \leq \psi^* \leq 1$, $|\cdot|$ and $\|\cdot\|^2$ are convex, we derive that

$$\begin{aligned} \frac{\rho_2}{2} (\|\mathbf{y}_e^0\|^2 + \|\tilde{\mathbf{y}}_e^0\|^2) &\geq \rho_1 \sum_{n=0}^N \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) \|\psi_e^{n+1}\|^2 + \sum_{n=0}^N \langle \tilde{\psi}_e^{n+1}, \psi_e^{n+1} \rangle \\ &+ \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle + \rho_2 \left(\frac{1}{2} \sum_{n=0}^N \|D\psi_e^{n+1} - \mathbf{y}_e^n\|^2 + \frac{1}{2} \|\mathbf{y}_e^{N+1}\|^2 \right). \end{aligned} \quad (64)$$

Firstly, (64) leads to

$$\sum_{n=0}^N \sum_{m=1}^2 (D_m u^*)^* (D_m u^*) \|\psi_e^{n+1}\|^2 < \infty, \quad (65)$$

which, together with Theorem 1, implies

$$\lim_{n \rightarrow \infty} \sum_{m=1}^2 (D_m u^n)^* (D_m u^n) \|\psi^n - \psi^*\|^2 = 0. \quad (66)$$

Denote the first term in (5) by

$$\mathcal{F}(\psi) = \mu \|Du^n\| \odot (\psi - 1) \odot \|\psi\|_2^2. \quad (67)$$

By the nonnegativity of the Bregman distance, we have $\lim_{n \rightarrow \infty} B_{\mathcal{F}}^{\mathbf{p}^*}(\psi^n, \psi^*) = 0$, i.e.,

$$\lim_{n \rightarrow \infty} \left(\mathcal{F}(\psi^n) - \mathcal{F}(\psi^*) - \langle Du^n \|^2 \odot (\psi^* - 1), \psi^n - \psi^* \rangle \right) = 0. \quad (68)$$

Secondly, (64) leads to

$$\sum_{n=0}^N \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle < +\infty, \text{ so } \lim_{n \rightarrow \infty} \langle \mathbf{p}_e^n, \mathbf{y}_e^{n+1} \rangle = 0. \quad (69)$$

Associating it with the nonnegativity of the Bregman distance (see [49,56]), we obtain

$$\lim_{n \rightarrow \infty} (|\mathbf{y}^n| - |\mathbf{y}^*| - \langle \mathbf{y}^n - \mathbf{y}^*, \mathbf{p}^* \rangle) = 0. \quad (70)$$

Thirdly, (64) leads also to $\sum_{n=0}^N \|D\psi_e^{n+1} - \mathbf{y}_e^n\|^2 < \infty$, which means

$$\lim_{n \rightarrow \infty} \|D\psi_e^{n+1} - \mathbf{y}_e^n\|^2 = 0. \quad (71)$$

By $D\psi^* = \mathbf{y}^*$, we have

$$\lim_{n \rightarrow \infty} \|D\psi^{n+1} - \mathbf{y}^n\| = 0. \quad (72)$$

Moreover, by the continuous property of $|\cdot|$, we obtain

$$\lim_{n \rightarrow \infty} (|D\psi^n| - |D\psi^*| - \langle D\psi^n - D\psi^*, \mathbf{p}^* \rangle) = 0. \quad (73)$$

Similarly, we have

$$\lim_{n \rightarrow \infty} (H_\psi(\psi^n) - H_\psi(\psi^*) - \langle \psi^n - \psi^*, \tilde{\psi}^* \rangle) = 0. \quad (74)$$

Combining (68), (73), (74) and (50), we have

$$\lim_{n \rightarrow \infty} (\mathcal{F}(\psi^n) + |D\psi^n| + H_\psi(\psi^n)) = \mathcal{F}(\psi^*) + |D\psi^*| + H_\psi(\psi^*). \quad (75)$$

This gives the first equation in (42). In the similar way as above, by denoting the first term in (9) be $\mathcal{G}(u)$, we have

$$\lim_{n \rightarrow \infty} (\mathcal{G}(u^n) + H_u(u^n)) = \mathcal{G}(u^*) + H_u(u^*), \quad (76)$$

where we have used the formula (52). This gives the second equation in (42).

Next, we prove the second equation in (43) by assuming that (9) has a unique solution. We prove that by contradiction. Assume that the second equation in (43) does not hold, which implies that there exists a subsequence u^{n_i} such that $\|u^{n_i} - u^*\| > \epsilon$ for some $\epsilon > 0$ and for all i . Let $c = tu^* + (1-t)u^{n_i}$ with $t \in (0, 1)$. By the convexity of $\mathfrak{R}$ and u^* is the unique minimizer of $\mathfrak{R}(u)$, we have

$$\mathfrak{R}(u^{n_i}) > t\mathfrak{R}(u^*) + (1-t)\mathfrak{R}(u^{n_i}) \geq \mathfrak{R}(c) \geq \min\{\mathfrak{R}(u) : \|u - u^*\| = \epsilon\}. \quad (77)$$

Denote

$$\bar{u} = \arg \min_u \{\mathfrak{R}(u) : \|u - u^*\| = \epsilon\}. \quad (78)$$

By applying the second equation in (43), we have

$$\mathfrak{R}(u^*) = \lim_{i \rightarrow \infty} \mathfrak{R}(u^{n_i}) \geq \mathfrak{R}(\bar{u}) > \mathfrak{R}(u^*), \quad (79)$$

which is a contradiction.

The first equation in (43) follows the same arguments as above.

References

- [1] B. Berkels, A. Rätz, M. Rumpf, A. Voigt, Extracting grain boundaries and macroscopic deformations from images on atomic scale, *J. Sci. Comput.* 35 (1) (2008) 1–23.
- [2] M. Upmannyu, R. Smith, D. Srolovitz, Atomistic simulation of curvature driven grain boundary migration, *Interface Sci.* 6 (1998) 41–58.
- [3] I. Sobel, An isotropic 3x3 image gradient operator, presentation at stanford A.I. project 1968, 2014.
- [4] L.G. Roberts, Machine Perception of Three-Dimensional Solids (ph.D thesis), Massachusetts Institute of Technology, 1963.
- [5] J.M. Prewitt, Object enhancement and extraction, *Pict. Process. Psychopictorics* 10 (1) (1970) 15–19.
- [6] J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. PAMI-8* (6) (1986) 679–698.
- [7] L. Alvarez, P.-L. Lions, J.-M. Morel, Image selective smoothing and edge detection by nonlinear diffusion. II, *SIAM J. Numer. Anal.* 29 (3) (1992) 845–866.
- [8] L.R. Liang, C.G. Looney, Competitive fuzzy edge detection, *Appl. Soft Comput.* 3 (2) (2003) 123–137.
- [9] B. Llanas, S. Lantarón, Edge detection by adaptive splitting, *J. Sci. Comput.* 46 (3) (2011) 486–518.
- [10] S. Mallat, W.L. Hwang, Singularity detection and processing with wavelets, *IEEE Trans. Inf. Theory* 38 (2) (1992) 617–643.
- [11] B. Rajathilagam, M. Rangarajan, Edge detection using G-lens based on matrix factorization by group representations, *Pattern Recognit.* 67 (2017) 1–15.
- [12] G. Wei, Y. Jia, Synchronization-based image edge detection, *Europhys. Lett.* 59 (6) (2002) 814–819.
- [13] F. Catté, P.-L. Lions, J.-M. Morel, T. Coll, Image selective smoothing and edge detection by nonlinear diffusion, *SIAM J. Numer. Anal.* 29 (1) (1992) 182–193.
- [14] P. Perona, J. Malik, Scale-space and edge-detection using anisotropic diffusion, *IEEE Trans. Pattern Anal.* 12 (7) (1990) 629–639.
- [15] P. Ganesan, G. Sajiv, A comprehensive study of edge detection for image processing applications, in: 2017 International Conference on Innovations in Information, Embedded and Communication Systems, ICIIECS, 2017, pp. 1–6.
- [16] D. Mumford, J. Shah, Optimal approximations by piecewise smooth functions and associated variational problems, *Comm. Pure Appl. Math.* 42 (5) (1989) 577–685.
- [17] L. Ambrosio, V. Tortorelli, Approximation of functions depending on jumps by elliptic functions via Γ -convergence, *Comm. Pure Appl. Math.* 13 (1990) 999–1036.
- [18] L. Ambrosio, V. Tortorelli, On the approximation of functionals depending on jumps by quadratic, elliptic functions, *Boll. Unione Mat. Ital.* 6 (1992) 105–123.
- [19] A. Blake, A. Zisserman, Visual reconstruction, *Math. Comp.* 373 (2) (1987) 131–135.
- [20] M. Zanetti, V. Ruggiero, M. Miranda Jr., Numerical minimization of a second-order functional for image segmentation, *Commun. Nonlinear Sci.* 36 (2016) 528–548.
- [21] L. Wang, Y. Shi, X. Tai, Robust edge detection using Mumford–Shah model and binary level set method, in: The Third International Conference on Scale Space and Variational Methods in Computer Vision, SSV2011, 2011, pp. 291–301.
- [22] Y. Shi, F. Guo, X. Su, J. Xu, Edge detection in presence of impulse noise, in: *Advances in Image and Graphics Technologies*, in: *Communications in Computer and Information Science*, vol. 437, Springer, 2014, pp. 8–18.
- [23] T. Chan, S. Esedoglu, M. Nikolova, Algorithms for finding global minimizers of image segmentation and denoising models, *SIAM J. Appl. Math.* 66 (5) (2006) 1632–1648.
- [24] S.I. Lee, H. Lee, P. Abbeel, A.Y. Ng, Efficient L_1 regularized logistic regression, in: National Conference on Artificial Intelligence, 2006, pp. 401–408.
- [25] Y. Shi, Y. Gu, L. Wang, X. Tai, A fast edge detection algorithm using binary labels, *Inverse Probl. Imag.* 9 (2) (2015) 551–578.
- [26] C. Revol-Muller, J.L. Rose, A. Pacureanu, F. Peyrin, C. Odet, Shape prior in variational region growing, in: 2012 3rd International Conference on Image Processing Theory, Tools and Applications, IPTA, 2012, pp. 116–120.
- [27] J.L. Rose, C. Revol-Muller, M. Almajdub, E. Chereul, C. Odet, Shape prior integrated in an automated 3D region growing method, in: IEEE International Conference on Image Processing, 2007, pp. 1 – 53–1 – 56.
- [28] C. Zotti, Z. Luo, O. Humbert, A. Lalande, P.M. Jodoin, GridNet with automatic shape prior registration for automatic MRI cardiac segmentation, in: International Workshop on Statistical Atlases and Computational Models of the Heart, 2017, pp. 73–81.
- [29] T. Riklin-Raviv, N. Kiryati, N. Sochen, Prior-based segmentation and shape registration in the presence of perspective distortion, *Int. J. Comput. Vis.* 72 (3) (2007) 309–328.
- [30] O. Gloger, K. Tönnies, Subject-specific prior shape knowledge in feature-oriented probability maps for fully automatized liver segmentation in MR volume data, *Pattern Recognit.* 84 (2018) 288–300.
- [31] G. Maicas, G. Carneiro, A.P. Bradley, Globally optimal breast mass segmentation from DCE-MRI using deep semantic segmentation as shape prior, in: IEEE International Symposium on Biomedical Imaging, 2017, pp. 305–309.
- [32] A. Tsai, A. Yezzi, W. Wells, C. Tempny, D. Tucker, A. Fan, W. Eric Grimson, A shape-based approach to the segmentation of medical imagery using level sets, *IEEE Trans. Med. Imaging* 22 (2) (2003) 137–154.
- [33] J. Kim, M. Cetin, A.S. Willsky, Nonparametric shape priors for active contour-based image segmentation, *Signal Process.* 87 (12) (2007) 3021–3044.

- [34] D. Cremers, S.J. Osher, S. Soatto, Kernel density estimation and intrinsic alignment, *Int. J. Comput. Vis.* 69 (3) (2004) 335–351.
- [35] H. Gao, O. Chae, Individual tooth segmentation from CT images using level set method with shape and intensity prior, *Pattern Recognit.* 43 (7) (2010) 2406–2417.
- [36] W. Guo, J. Qin, S. Tari, Automatic Prior Shape Selection for Image Segmentation, Springer International Publishing, 2015, pp. 1–8.
- [37] Y. Wang, J. Yang, W. Yin, Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, *SIAM J. Imaging Sci.* 1 (3) (2008) 248–272.
- [38] C.R. Vogel, M.E. Oman, Iterative methods for total variation denoising, *SIAM J. Sci. Comput.* 17 (1) (1996) 227–238.
- [39] Y. Shi, Q. Chang, Acceleration methods for image restoration problem with different boundary conditions, *Appl. Numer. Math.* 58 (5) (2008) 602–614.
- [40] N. Badshah, K. Chen, On two multigrid algorithms for modeling variational multiphase image segmentation, *IEEE Trans. Image Process.* 18 (5) (2009) 1097–1106.
- [41] R. Glowinski, A. Marrocco, Sur l'approximation, par éléments finis d'ordre 1, et la résolution, par pénalisation-dualité, d'une classe de problèmes de Dirichlet non linéaires., *J. Equine Vet. Sci.* 2 (R-2) (1975) 41–76.
- [42] X. Bresson, T. Chan, Fast dual minimization of the vectorial total variation norm and applications to color image processing, *Inverse Probl. Imag.* 2 (4) (2008) 455–484.
- [43] M.A.T. Figueiredo, R.D. Nowak, S.J. Wright, Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, in: *IEEE Journal of Selected Topics in Signal Processing*, 2007, pp. 586–597.
- [44] J.L. Mohammed, R.A. Hummel, S.W. Zucker, A Gradient Projection Algorithm for Relaxation Methods, *IEEE Computer Society*, 1983, pp. 330–332.
- [45] T.F. Chan, T.P. Mathew, Domain decomposition algorithms, *Acta Numer.* 3 (3) (1994) 61–143.
- [46] A. Toselli, O.B. Widlund, Domain Decomposition Methods Algorithms and Theory, Vol. 34, Springer, 2005, p. xvi,450.
- [47] M. Nikolova, M. Ng, Fast image reconstruction algorithms combining half-quadratic regularization and preconditioning, in: *Image Processing, 2001. Proceedings. 2001 International Conference on*, Vol. 1, IEEE, 2001, pp. 277–280.
- [48] R.H. Chan, M. Tao, X. Yuan, Constrained total variation deblurring models and fast algorithms based on alternating direction method of multipliers, *SIAM J. Imaging Sci.* 6 (1) (2013) 680–697.
- [49] J.-F. Cai, S. Osher, Z. Shen, Split Bregman methods and frame based image restoration, *Multiscale Model. Simul.* 8 (2) (2009) 337–369.
- [50] C. Wu, X. Tai, Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models, *SIAM J. Imaging Sci.* 3 (2010) 300–339.
- [51] Y. Shi, L. Wang, X. Tai, Geometry of total variation regularized L_p -model, *J. Comput. Appl. Math.* 236 (8) (2012) 2223–2234.
- [52] X. Tai, C. Wu, Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, in: *Scale Space and Variational Methods in Computer Vision, Proceedings*, in: *Lecture Notes in Computer Science*, vol. 5567, Elsevier, 2009, pp. 502–513.
- [53] D. Gabay, B. Mercier, A dual algorithm for the solution of nonlinear variational problems via finite element approximation, *Comput. Math. Appl.* 2 (1) (1976) 17–40.
- [54] A. Chambolle, An algorithm for total variation minimization and applications, *J. Math. Imaging Vision* 20 (1–2) (2004) 89–97.
- [55] L. Wang, Y. You, H. Lian, Convergence and sparsity of Lasso and group Lasso in high-dimensional generalized linear models, *Stat. Pap.* 56 (3) (2015) 819–828.
- [56] S. Osher, M. Burger, D. Goldfarb, J. Xu, W. Yin, An iterative regularization method for total variation-based image restoration, *Multiscale Model. Simul.* 4 (2) (2005) 460–489.