Fast Algorithms for p-elastica Energy with the Application to Image Inpainting and Curve Reconstruction*

Jooyoung Hahn[1,**], Ginmo J. Chung[2], Yu Wang[2], and Xue-Cheng Tai[2,3]

[1] Institute for Mathematics and Scientific Computing, University of Graz, Austria
[2] Division of Mathematical Sciences, School of Physical Mathematical Sciences, Nanyang Technological University, Singapore
[3] Mathematics Institute, University of Bergen, Norway
joo.hahn@uni-graz.at

Abstract. In this paper, we propose fast and efficient algorithms for p-elastica energy ($p = 1$ or 2). Inspired by the recent algorithm for Euler's elastica models in [16], the algorithm is extended to solve the problem related to p-elastica energy based on augmented Lagrangian method. The proposed algorithms are as efficient as the previous method in terms of low computational cost per iteration. We provide an algorithm which replaces fast Fourier transform (FFT) by a cheap arithmetic operation at each grid point. Numerical tests on image inpainting are provided to demonstrate the efficiency of the proposed algorithms. We also show examples of using the proposed algorithms in curve reconstruction from unorganized data set.

Keywords: p-elastica energy, Augmented Lagrangian method, Euler's elasitca, Image inpainting, Curve reconstruction, Unorganized data set.

1 Introduction

The curvature of the curve has been extensively used in minimization problems in image processing and computer vision. D. Mumford, M. Nitzberg, and T. Shiota [14] introduced segmentation with depth to find a continuation curve γ which minimizes Euler's elastica energy ($p = 2$):

$$\mathcal{E}(\gamma) = \int_{\gamma} (a + b|\kappa|^p)\, ds, \tag{1}$$

* The research is supported by MOE (Ministry of Education) Tier II project T207N2202 and IDM project NRF2007IDMIDM002-010. In addition, the support from SUG 20/07 is also gratefully acknowledged.

** This author is currently at Institute of Mathematics and Scientific Computing in University of Graz, Austria. He has been supported by the Austrian Science Fund (FWF) under the START-Program Y305 "Interfaces and Free Boundaries" and the SFB "Mathematical Optimization and Its Applications in Biomedical Sciences" since November 2010.

A.M. Bruckstein et al. (Eds.): SSVM 2011, LNCS 6667, pp. 169–182, 2012.

where κ is the curvature of the curve in $\mathbf{R}^2$, a and b are positive constants, and $p \geq 1$. In [2], Euler's elastica problem is reduced to solve a set of algebraic equations in Jacobi's functions. Semicontinuity and relaxation properties of (1) were presented in [6]. Following the work [14], Masnou and Morel [13] proposed a variational formulation in the geometrical recovery of the missing parts from a given image $u_0 : \mathcal{D} \setminus \tilde{\mathcal{D}} \subset \mathbf{R}^2 \to \mathbf{R}$, where $\mathcal{D} \supset \tilde{\mathcal{D}}$. In [13], an energy functional to complete of all level lines of u_0 is written by using change of variable and the coarea formula from (1):

$$\int_{\mathcal{D}} \left(a + b \left| \nabla \cdot \frac{\nabla u}{|\nabla u|} \right|^p \right) |\nabla u|, \tag{2}$$

where $p \geq 1$. Note that the standard Lebesgue measure in $\mathbf{R}^2$ is omitted in the rest of paper. The authors in [8] solved the minimization problem (2) with $p = 2$ by using the Euler-Lagrange equation and the gradient descent method. In [9], they showed that the curvature term is essential to achieve a connectivity principle. The properties of variational model and the existence of minimizing functional (2) are investigated by Ambrosio and Masnou [1]. The authors in [4] proposed an energy functional minimization with two arguments, $\mathbf{n}$ which represents the normalized image gradient and a gray image (real-valued function) u defined on $\mathcal{D}$:

$$\begin{aligned} &\min_{\mathbf{n},u} \left(\int_{\mathcal{D}} |\nabla \cdot \mathbf{n}|^p \left(c_1 + c_2 |\nabla k * u| \right) + \zeta \int_{\mathcal{D}} \left(|\nabla u| - \mathbf{n} \cdot \nabla u \right) \right), \\ &|\mathbf{n}| \leq 1, \quad ||u|| \leq ||u_0||_{L^\infty(\mathcal{D} \setminus \tilde{\mathcal{D}})}, \end{aligned} \tag{3}$$

where c_1 and c_2 are positive constants, k denotes a Gaussian kernel; see [4] for boundary conditions and the detail admissible sets. Note that the constraint term $|\nabla u| - \mathbf{n} \cdot \nabla u$ in (3) is crucially used in [16]. The existence of minimizers of a relaxed variant of (3) is proved in [5].

Efficient numerical algorithms for energy minimization related to the curvature are studied very recently. The authors [15] used a linear programming relaxation and the discrete elastica [7] to minimize energy functionals for image segmentation and inpainting with curvature regularity. The algorithm in [15] is independent of initialization and computes the global minimum. An improved fast algorithm to the elastica model in [15] is introduced in [10]. The authors in [3] proposed an efficient algorithm based on graph cuts for minimizing the Euler's elastica model for image denoising and inpainting. In [16], new variables and several constraint conditions are introduced to change the Euler's elastica model into a constraint minimization and then augmented Lagrangian method (ALM) is used to obtain a stationary point.

In this paper, we present fast and efficient algorithms for p-elastica energy:

$$\int_{\Omega} \left(a + b \left| \nabla \cdot \frac{\nabla u}{|\nabla u|} \right|^p \right) |\nabla u| + \frac{\eta}{q} \int_{\Gamma} |u - u_0|^q, \tag{4}$$

where $p \geq 1$, $q \geq 1$, Ω is the domain of image u, and $\Gamma \subsetneq \Omega$ is the domain of a given image u_0. The minimization of the functional (4) interpolates the

values u_0 on the boundary $\partial\Gamma$ into the inpainting domain $\Omega \setminus \Gamma$. Inspired by the recent algorithm in [16], we extend the algorithm to minimize the p-elastica energy (4). The proposed algorithms use less memory and lower computational cost per iteration than [16]. Numerical tests on image inpainting are provided to demonstrate the efficiency of the proposed algorithms. Moreover, we present a model and numerical examples for curve reconstruction from unorganized data set which has the same regularity term in (4) with $p = 1$.

2 Review of ALM for Euler's Elastica Model

In this section, the augmented Lagrangian method for Euler's elastica model [16] is briefly introduced and we discuss properties of the algorithm and possible improvements in terms of computational cost. When $p = 2$ in (4), the authors [16] proposed several new variables to change the energy minimization of (4) into the constraint minimization problem:

$$\begin{aligned} \min_{v,u,\mathbf{m},\mathbf{p},\mathbf{n}} & \int_\Omega \left(a + b(\nabla \cdot \mathbf{n})^2\right) |\mathbf{p}| + \frac{\eta}{q} \int_\Gamma |v - u_0|^q \\ & \text{with} \quad v = u, \quad \mathbf{p} = \nabla u, \quad \mathbf{n} = \mathbf{m}, \quad |\mathbf{p}| = \mathbf{m} \cdot \mathbf{p}, \quad |\mathbf{m}| \le 1. \end{aligned} \tag{5}$$

Note that the variable $\mathbf{m}$ plays an important role to avoid nonuniqueness of a solution in the Euler-Lagrange equation for $\mathbf{n}$-subproblem; see more details in [16]. In order to solve the constraint optimization problem (5), the following augmented Lagrangian functional is used:

$$\begin{aligned} \mathcal{L}(v, u, \mathbf{m}, \mathbf{p}, \mathbf{n}; \lambda_1, \boldsymbol{\lambda}_2, \lambda_3, \boldsymbol{\lambda}_4) &= \int_\Omega \left(a + b(\nabla \cdot \mathbf{n})^2\right) |\mathbf{p}| + \frac{\eta}{q} \int_\Gamma |v - u_0|^q \\ &+ r_1 \int_\Omega (|\mathbf{p}| - \mathbf{m} \cdot \mathbf{p}) + \int_\Omega \lambda_1 (|\mathbf{p}| - \mathbf{m} \cdot \mathbf{p}) + \frac{r_2}{2} \int_\Omega |\mathbf{p} - \nabla u|^2 \\ &+ \int_\Omega \boldsymbol{\lambda}_2 \cdot (\mathbf{p} - \nabla u) + \frac{r_3}{2} \int_\Omega (v - u)^2 + \int_\Omega \lambda_3 (v - u) \\ &+ \frac{r_4}{2} \int_\Omega |\mathbf{n} - \mathbf{m}|^2 + \int_\Omega \boldsymbol{\lambda}_4 \cdot (\mathbf{n} - \mathbf{m}) + \delta_{\mathcal{R}}(\mathbf{m}), \end{aligned} \tag{6}$$

where λ_1, $\boldsymbol{\lambda}_2$, λ_3, and $\boldsymbol{\lambda}_4$ are Lagrange multipliers, r_1, r_2, r_3, and r_4 are positive penalty parameters, and an indicator function $\delta_{\mathcal{R}}(\cdot)$ on $\mathcal{R} = \{\mathbf{m} \in \mathbf{L}^2(\Omega) \mid |\mathbf{m}| \le 1 \text{ a.e. in } \Omega\}$ is defined by

$$\delta_{\mathcal{R}}(\mathbf{m}) = \begin{cases} 0 & \mathbf{m} \in \mathcal{R}, \\ +\infty & \text{otherwise.} \end{cases}$$

Note that the constraint $|\mathbf{m}| \le 1$ is imposed by the indicator function and then we have $|\mathbf{p}| - \mathbf{m} \cdot \mathbf{p} \ge 0$, a.e. in Ω. That is, it is not necessarily to use L^2 penalization for the term multiplied by r_1 which causes nonlinearity in the $\mathbf{p}$-subproblem.

An iterative algorithm is suggested to find a stationary point of (6). Lagrange multipliers λ_1^0, $\boldsymbol{\lambda}_2^0$, λ_3^0, and $\boldsymbol{\lambda}_4^0$ and the variables v^0, u^0, $\mathbf{m}^0$, $\mathbf{p}^0$, and $\mathbf{n}^0$ are initialized to zero. For $k \geq 0$, an approximate minimizer

$$\left(v^{k+1}, u^{k+1}, \mathbf{m}^{k+1}, \mathbf{p}^{k+1}, \mathbf{n}^{k+1}\right) \simeq \underset{v,u,\mathbf{m},\mathbf{p},\mathbf{n}}{\arg\min}\, \mathcal{L}(v, u, \mathbf{m}, \mathbf{p}, \mathbf{n}; \lambda_1^k, \boldsymbol{\lambda}_2^k, \lambda_3^k, \boldsymbol{\lambda}_4^k)$$

is obtained by alternatingly solving the subproblems. Let $\tilde{v}^0 = v^k$, $\tilde{u}^0 = u^k$, $\tilde{\mathbf{m}}^0 = \mathbf{m}^k$, $\tilde{\mathbf{p}}^0 = \mathbf{p}^k$, and $\tilde{\mathbf{n}}^0 = \mathbf{n}^k$. For $l = 0, \cdots, L-1$, minimizers $\tilde{v}^{l+1}$, $\tilde{u}^{l+1}$, $\tilde{\mathbf{m}}^{l+1}$, $\tilde{\mathbf{p}}^{l+1}$, and $\tilde{\mathbf{n}}^{l+1}$ are approximately obtained by alternatingly minimizing the following energy functionals:

$$\mathcal{E}_1(v) = \frac{\eta}{q}\int_\Gamma |v - u_0|^q + \int_\Omega \frac{r_3}{2}\left(v - \tilde{u}^l\right)^2 + \lambda_3^k v, \tag{7}$$

$$\mathcal{E}_2(u) = \int_\Omega \frac{r_2}{2}\left|\tilde{\mathbf{p}}^l - \nabla u\right|^2 - \boldsymbol{\lambda}_2^k \cdot \nabla u + \frac{r_3}{2}\left(\tilde{v}^{l+1} - u\right)^2 + \lambda_3^k(-u), \tag{8}$$

$$\mathcal{E}_3(\mathbf{m}) = \delta_{\mathcal{R}}(\mathbf{m}) + \int_\Omega \frac{r_4}{2}\left|\tilde{\mathbf{n}}^l - \mathbf{m}\right|^2 - \boldsymbol{\lambda}_4^k \cdot \mathbf{m} - (r_1 + \lambda_1^k)\mathbf{m} \cdot \tilde{\mathbf{p}}^l, \tag{9}$$

$$\begin{aligned}\mathcal{E}_4(\mathbf{p}) = &\int_\Omega \left(a + b\left(\nabla \cdot \tilde{\mathbf{n}}^l\right)^2\right)|\mathbf{p}| + (r_1 + \lambda_1^k)\left(|\mathbf{p}| - \tilde{\mathbf{m}}^{l+1} \cdot \mathbf{p}\right) \\ &+ \int_\Omega \frac{r_2}{2}\left|\mathbf{p} - \nabla \tilde{u}^{l+1}\right|^2 + \boldsymbol{\lambda}_2^k \cdot \mathbf{p},\end{aligned} \tag{10}$$

$$\mathcal{E}_5(\mathbf{n}) = \int_\Omega b(\nabla \cdot \mathbf{n})^2\left|\tilde{\mathbf{p}}^{l+1}\right| + \frac{r_4}{2}\left|\mathbf{n} - \tilde{\mathbf{m}}^{l+1}\right|^2 + \boldsymbol{\lambda}_4^k \cdot \mathbf{n}. \tag{11}$$

After L iterations, variables at $(k+1)^{\text{th}}$ step are updated:

$$\left(v^{k+1}, u^{k+1}, \mathbf{m}^{k+1}, \mathbf{p}^{k+1}, \mathbf{n}^{k+1}\right) = \left(\tilde{v}^L, \tilde{u}^L, \tilde{\mathbf{m}}^L, \tilde{\mathbf{p}}^L, \tilde{\mathbf{n}}^L\right).$$

A large number of iteration L may be necessary to find the minimizers of the above functional $\mathcal{L}(v, u, \mathbf{m}, \mathbf{p}, \mathbf{n}; \lambda_1^k, \boldsymbol{\lambda}_2^k, \lambda_3^k, \boldsymbol{\lambda}_4^k)$. However, it is empirically enough to use $L = 1$ according to recent literatures [17,19]. Lagrange multipliers λ_1^{k+1}, $\boldsymbol{\lambda}_2^{k+1}$, λ_3^{k+1}, and $\boldsymbol{\lambda}_4^{k+1}$ are updated by the standard method in augmented Lagrangian method; see details in [16]. For $q = 1$ or 2, subproblems to minimize $\mathcal{E}_1(v)$, $\mathcal{E}_3(\mathbf{m})$, and $\mathcal{E}_4(\mathbf{p})$ can be solved by closed form formulas which only take arithmetic operations at each grid point. Subproblems to minimize $\mathcal{E}_2(u)$ and $\mathcal{E}_5(\mathbf{n})$ need to solve a linear partial differential equation (PDE) and a coupled PDE with variable coefficients, respectively.

The most difficult and time-consuming process in the algorithm [16] is to solve the Euler-Lagrange equation of (11):

$$-2\nabla\left(b\left|\tilde{\mathbf{p}}^{l+1}\right|\nabla \cdot \mathbf{n}\right) + r_4\left(\mathbf{n} - \tilde{\mathbf{m}}^{l+1}\right) + \boldsymbol{\lambda}_4^k = 0. \tag{12}$$

A frozen coefficient method with FFT is suggested to solve (12) and it needs an inner iteration. Since the equation is a coupled PDE, FFT is used twice per

each inner iteration; see details in [16]. Since FFT is also used to solve the Euler-Lagrange equation of (8), the algorithm needs FFT more than three times for each outer iteration k. Therefore, if an algorithm uses FFT three times per outer iteration, it will be optimal. Such an optimality can be achieved as long as the variable coefficient $b|\mathbf{p}^{l+1}|$ in (12) is eliminated.

When the boundary condition is directly imposed without using the fidelity term $\frac{\eta}{q}\int_\Gamma |u-u_0|^q$ in (5) and the irregular inpainting domain is assigned, it is obvious that FFT cannot be used to solve the Euler-Lagrange equations of (8) and (11). Moreover, there are also some applications in [12] which we cannot use FFT in the augmented Lagrangian method [17]. In the case of u-subproblem (8), even though FFT is not used, it does not make any difficulties to find the minimizer because the Euler-Lagrange equation of (8) is a linear PDE:

$$-r_2\triangle u + r_3 u = r_3\tilde{v}^{l+1} + \lambda_3^k - r_2\nabla\cdot\tilde{\mathbf{p}}^l - \nabla\cdot\boldsymbol{\lambda}_2^k.$$

We can use linear iterative methods for symmetric positive definite matrix. However, it is not straightforward to solve the equation (12) in this manner. Considering some applications defined on a two dimensional surface in $\mathbf{R}^3$, it is necessary to develop an algorithm to efficiently minimize p-elastica model (4) without using FFT.

Note that one may use the gradient descent method to find a minimizer of (11). If the explicit method is applied, a small time step should be used because of large variation of $|\tilde{\mathbf{p}}^{l+1}|$ in the domain and then it makes a slow convergence. Since the coefficient $b\left|\tilde{\mathbf{p}}^{l+1}\right|$ in (12) varies in the domain and the equations are coupled, the implicit method is difficult to be applied.

3 Proposed Algorithms

In this section, we propose two algorithms. The first method is designed to eliminate the variable coefficients in (12) and then an optimal number of using FFT is achieved for each outer iteration. The second method replaces FFT procedure in [16] for minimizing (8) and (17) into a very simple updating scheme and it is memory efficient comparing with the first method. Both algorithms are as fast as the algorithm in Section 2.

3.1 Method 1

In order to extend the algorithm [16] to solve the p-elastica problem ($p = 1$ or 2) and remove the variable coefficient in (12), we simply introduce a new variable

$$g = \nabla\cdot\mathbf{n}. \tag{13}$$

That is, we use the augmented Lagrangian functional for p-elastica problem with the additional positive penalty parameter r_5 and the Lagrange multiplier λ_5:

$$\begin{aligned}\mathcal{L}^1(v,u,\mathbf{m},\mathbf{p},g,\mathbf{n};\lambda_1,\boldsymbol{\lambda}_2,\lambda_3,\boldsymbol{\lambda}_4,\lambda_5) &= \int_\Omega (a+b|g|^p)\,|\mathbf{p}| + \frac{\eta}{q}\int_\Gamma |v-u_0|^q \\ &+ r_1\int_\Omega (|\mathbf{p}|-\mathbf{m}\cdot\mathbf{p}) + \int_\Omega \lambda_1(|\mathbf{p}|-\mathbf{m}\cdot\mathbf{p}) + \frac{r_2}{2}\int_\Omega |\mathbf{p}-\nabla u|^2 + \delta_{\mathcal{R}}(\mathbf{m}) \\ &+ \int_\Omega \boldsymbol{\lambda}_2\cdot(\mathbf{p}-\nabla u) + \frac{r_3}{2}\int_\Omega (v-u)^2 + \int_\Omega \lambda_3(v-u) + \frac{r_4}{2}\int_\Omega |\mathbf{n}-\mathbf{m}|^2 \\ &+ \int_\Omega \boldsymbol{\lambda}_4\cdot(\mathbf{n}-\mathbf{m}) + \frac{r_5}{2}\int_\Omega (\nabla\cdot\mathbf{n}-g)^2 + \int_\Omega \lambda_5(\nabla\cdot\mathbf{n}-g).\end{aligned} \tag{14}$$

We use the same iterative algorithm for (6) to find a stationary point of (14). After all variables and Lagrange multipliers are initialized to zero, an approximate minimizer for $k \geq 0$

$$\begin{aligned}&\left(v^{k+1},u^{k+1},\mathbf{m}^{k+1},\mathbf{p}^{k+1},g^{k+1},\mathbf{n}^{k+1}\right) \\ &\simeq \underset{v,u,\mathbf{m},\mathbf{p},g,\mathbf{n}}{\arg\min}\ \mathcal{L}^1(v,u,\mathbf{m},\mathbf{p},g,\mathbf{n};\lambda_1^k,\boldsymbol{\lambda}_2^k,\lambda_3^k,\boldsymbol{\lambda}_4^k,\lambda_5^k)\end{aligned}$$

is obtained by alternatingly solving the subproblems. Letting $\tilde{v}^0 = v^k$, $\tilde{u}^0 = u^k$, $\tilde{\mathbf{m}}^0 = \mathbf{m}^k$, $\tilde{\mathbf{p}}^0 = \mathbf{p}^k$, $\tilde{g}^0 = g^k$, and $\tilde{\mathbf{n}}^0 = \mathbf{n}^k$, for $l = 0,\cdots,L-1$, we find minimizers $\tilde{v}^{l+1}$, $\tilde{u}^{l+1}$, $\tilde{\mathbf{m}}^{l+1}$, $\tilde{\mathbf{p}}^{l+1}$, $\tilde{g}^{l+1}$, and $\tilde{\mathbf{n}}^{l+1}$ of the following energy functionals:

$$\mathcal{E}_1^1(v) = \mathcal{E}_1(v), \quad \mathcal{E}_2^1(u) = \mathcal{E}_2(u), \quad \mathcal{E}_3^1(\mathbf{m}) = \mathcal{E}_3(\mathbf{m}),$$

$$\begin{aligned}\mathcal{E}_4^1(\mathbf{p}) &= \int_\Omega \left(a+b\left|\tilde{g}^l\right|^p\right)|\mathbf{p}| + (r_1+\lambda_1^k)\left(|\mathbf{p}|-\tilde{\mathbf{m}}^{l+1}\cdot\mathbf{p}\right) \\ &\quad + \int_\Omega \frac{r_2}{2}\left|\mathbf{p}-\nabla\tilde{u}^{l+1}\right|^2 + \boldsymbol{\lambda}_2^k\cdot\mathbf{p},\end{aligned} \tag{15}$$

$$\mathcal{E}_5^1(g) = \int_\Omega b\left|\tilde{\mathbf{p}}^{l+1}\right||g|^p + \frac{r_5}{2}\left(\nabla\cdot\tilde{\mathbf{n}}^l - g\right)^2 + \lambda_5^k(-g), \tag{16}$$

$$\mathcal{E}_6^1(\mathbf{n}) = \int_\Omega \frac{r_4}{2}\left|\mathbf{n}-\tilde{\mathbf{m}}^{l+1}\right|^2 + \boldsymbol{\lambda}_4^k\cdot\mathbf{n} + \frac{r_5}{2}\left(\nabla\cdot\mathbf{n}-\tilde{g}^{l+1}\right)^2 + \lambda_5^k\nabla\cdot\mathbf{n}. \tag{17}$$

After L iterations, we update

$$\left(v^{k+1},u^{k+1},\mathbf{m}^{k+1},\mathbf{p}^{k+1},g^{k+1},\mathbf{n}^{k+1}\right) = \left(\tilde{v}^L,\tilde{u}^L,\tilde{\mathbf{m}}^L,\tilde{\mathbf{p}}^L,\tilde{g}^L,\tilde{\mathbf{n}}^L\right).$$

In practice, $L = 1$ is used to make a consistent and fair comparison with the algorithm in Section 2. For p and $q = 1$ or 2, there are closed form formulas to find minimizers of $\mathcal{E}_1^1(v)$, $\mathcal{E}_3^1(\mathbf{m})$, $\mathcal{E}_4^1(\mathbf{p})$, and $\mathcal{E}_5^1(g)$ and it takes simply arithmetic operations at each grid point. We use FFT to solve the Euler's Lagrange equation of $\mathcal{E}_2^1(u)$.

Now, the Euler-Lagrange equation of $\mathcal{E}_6^1(\mathbf{n})$ is a linear coupled PDE:

$$-r_5\nabla(\nabla\cdot\mathbf{n}) + r_4\mathbf{n} = r_4\tilde{\mathbf{m}}^{l+1} - \boldsymbol{\lambda}_4^k - r_5\nabla\tilde{g}^{l+1} + \nabla\lambda_5^k. \tag{18}$$

The variable coefficient in (12) is removed and solution $\tilde{\mathbf{n}}^{l+1}$ of PDE are unique with a suitable boundary condition and the positive penalty parameter r_4. In

the discrete frequency domain, the coupled PDE (18) yields a 2 by 2 system of equation for each frequency and the determinant of the coefficient matrix is not zero if $r_4 > 0$. Note that the operator $\nabla(\nabla\cdot)$ is singular and then the coupled PDE becomes unstable if $r_4 \simeq 0$. Unlike the iterative method for solving (12) in [16], the coupled PDE (18) can be directly solved with just two FFT algorithms to obtain a minimizer of $\mathcal{E}_6^1(\mathbf{n})$.

One may use the gradient descent method to find the minimizer of $\mathcal{E}_6^1(\mathbf{n})$ (17). However, it introduces another variable for time step which should be properly chosen depending on r_4 and r_5.

In numerical examples of Method 1, we use FFT for subproblems $\mathcal{E}_2^1(u)$ and $\mathcal{E}_6^1(\mathbf{n})$ to make a fair comparison with the algorithm in Section 2.

3.2 Method 2

In this subsection, we propose an algorithm which is more effective than previous algorithms in terms of using memory and is as fast as the Method 1 in subsection 3.1 and the algorithm in Section 2. Note that the algorithms for (6) and (14) need to use 14 and 16 arrays, respectively, which have the same size as a given image u_0. The easiest technique to reduce memory usage is to eliminate unnecessary variables. Even though all variables in (14) play an important role in separating nonlinear properties and dissolving higher order derivatives in p-elastica model (4), the variable $\mathbf{m}$ may not be very crucial because $|\mathbf{n}| \leq 1$ can be achieved by a brute force method. Therefore, we simply propose the following augmented Lagrangian functional for p-elastica problem:

$$\begin{aligned}\mathcal{L}^2\left(v,u,\mathbf{p},g,\mathbf{n};\mu_1,\boldsymbol{\mu}_2,\mu_3,\mu_4\right) &= \int_\Omega \left(a+b|g|^p\right)|\mathbf{p}| + \frac{\eta}{q}\int_\Gamma |v-u_0|^q \\ &+ c_1\int_\Omega (|\mathbf{p}| - \mathbf{n}\cdot\mathbf{p}) + \int_\Omega \mu_1(|\mathbf{p}| - \mathbf{n}\cdot\mathbf{p}) + \frac{c_2}{2}\int_\Omega |\mathbf{p}-\nabla u|^2 \\ &+ \int_\Omega \boldsymbol{\mu}_2\cdot(\mathbf{p}-\nabla u) + \frac{c_3}{2}\int_\Omega (v-u)^2 + \int_\Omega \mu_3(v-u) \\ &+ \frac{c_4}{2}\int_\Omega (\nabla\cdot\mathbf{n}-g)^2 + \int_\Omega \mu_4(\nabla\cdot\mathbf{n}-g), \quad \text{with} \quad |\mathbf{n}|\leq 1,\end{aligned} \tag{19}$$

where μ_1, $\boldsymbol{\mu}_2$, μ_3, and μ_4 are Lagrange multipliers, c_1, c_2, c_3, and c_4 are positive penalty parameters. Note that the algorithm for $\mathcal{L}^2$ needs 12 arrays which have the same size as a given image u_0.

We use the same iterative algorithm for (6) to find a stationary point of (19). After all variables and Lagrange multipliers are initialized to zero, for $k \geq 0$, an approximate minimizer

$$\left(v^{k+1},u^{k+1},\mathbf{p}^{k+1},g^{k+1},\mathbf{n}^{k+1}\right) \simeq \underset{v,u,\mathbf{m},\mathbf{p},g,\mathbf{n}}{\arg\min}\ \mathcal{L}^2(v,u,\mathbf{m},\mathbf{p},g,\mathbf{n};\mu_1^k,\boldsymbol{\mu}_2^k,\mu_3^k,\mu_4^k)$$

is obtained by alternatingly solving the subproblems. Letting $\tilde{v}^0 = v^k$, $\tilde{u}^0 = u^k$, $\tilde{\mathbf{p}}^0 = \mathbf{p}^k$, $\tilde{g}^0 = g^k$, and $\tilde{\mathbf{n}}^0 = \mathbf{n}^k$, for $l = 0, \cdots, L-1$, we find minimizers $\tilde{v}^{l+1}$, $\tilde{u}^{l+1}$, $\tilde{\mathbf{p}}^{l+1}$, $\tilde{g}^{l+1}$, and $\tilde{\mathbf{n}}^{l+1}$ of the following energy functionals:

$$\mathcal{E}_1^2(v) = \frac{\eta}{q}\int_\Gamma |v - u_0|^q + \int_\Omega \frac{c_3}{2}\left(v - \tilde{u}^l\right)^2 + \mu_3^k v, \tag{20}$$

$$\mathcal{E}_2^2(u) = \int_\Omega \frac{c_2}{2}\left|\tilde{\mathbf{p}}^l - \nabla u\right|^2 - \boldsymbol{\mu}_2^k \cdot \nabla u + \frac{c_3}{2}\left(\tilde{v}^{l+1} - u\right)^2 + \mu_3^k(-u), \tag{21}$$

$$\begin{aligned}\mathcal{E}_3^2(\mathbf{p}) = &\int_\Omega \left(a + b\left|\tilde{g}^l\right|^p\right)|\mathbf{p}| + (c_1 + \mu_1^k)\left(|\mathbf{p}| - \tilde{\mathbf{n}}^{l+1}\cdot\mathbf{p}\right), \\ &+ \int_\Omega \frac{c_2}{2}\left|\mathbf{p} - \nabla\tilde{u}^{l+1}\right|^2 + \boldsymbol{\mu}_2^k \cdot \mathbf{p},\end{aligned} \tag{22}$$

$$\mathcal{E}_4^2(g) = \int_\Omega b\left|\tilde{\mathbf{p}}^{l+1}\right||g|^p + \frac{c_4}{2}\left(\nabla\cdot\tilde{\mathbf{n}}^l - g\right)^2 + \mu_4^k(-g), \tag{23}$$

$$\mathcal{E}_5^2(\mathbf{n}) = \int_\Omega \frac{c_4}{2}\left(\nabla\cdot\mathbf{n} - \tilde{g}^{l+1}\right)^2 + \mu_4^k\nabla\cdot\mathbf{n} - (c_1+\mu_1)\,\mathbf{n}\cdot\tilde{\mathbf{p}}^{l+1}. \tag{24}$$

After L iterations, we update

$$\left(v^{k+1}, u^{k+1}, \mathbf{p}^{k+1}, g^{k+1}, \mathbf{n}^{k+1}\right) = \left(\tilde{v}^L, \tilde{u}^L, \tilde{\mathbf{p}}^L, \tilde{g}^L, \tilde{\mathbf{n}}^L\right).$$

Similar to the Method 1, we observe that $L > 1$ does not make quite different numerical results from using $L = 1$.

Now, one may easily notice that we have a huge problem in the $\mathbf{n}$-subproblem for minimizing the functional (24) whose the Euler-Lagrange equation is

$$-c_4\nabla\left(\nabla\cdot\mathbf{n} - \tilde{g}^{l+1}\right) - (c_1+\mu_1)\,\tilde{\mathbf{p}}^{l+1} - \nabla\mu_4^k = 0. \tag{25}$$

Obviously, the solution of PDE is not unique because of the operator $\nabla(\nabla\cdot)$. Comparing with (12) and (18), such a problem is caused by the lack of linear term in the coupled PDE (25), which has been provided by the new variable $\mathbf{m}$ in the augmented Lagrangian functionals in (6) and (14). However, we simply generate a linear term by the linearization of L^2 penalization for $g = \nabla\cdot\mathbf{n}$, inspired by the linearized proximal alternating minimization algorithm in [20]. That is, the energy functional (24) can be approximated by linearization of $(\nabla\cdot\mathbf{n} - g)^2$ at $\tilde{\mathbf{n}}^l$:

$$\begin{aligned}\mathcal{E}_5^2(\mathbf{n}) &= \int_\Omega \mu_4^k\nabla\cdot\mathbf{n} - (c_1+\mu_1)\,\mathbf{n}\cdot\tilde{\mathbf{p}}^{l+1} + \frac{c_4}{2}\left(\nabla\cdot\mathbf{n} - \tilde{g}^{l+1}\right)^2 \\ &\simeq \int_\Omega \mu_4^k\nabla\cdot\mathbf{n} - (c_1+\mu_1)\,\mathbf{n}\cdot\tilde{\mathbf{p}}^{l+1} \\ &\quad + \int_\Omega \frac{c_4}{2}\left(\left(\nabla\cdot\tilde{\mathbf{n}}^l - \tilde{g}^{l+1}\right)^2 - 2\nabla\left(\nabla\cdot\tilde{\mathbf{n}}^l - \tilde{g}^{l+1}\right)\cdot\left(\mathbf{n} - \tilde{\mathbf{n}}^l\right) + \delta\left|\mathbf{n} - \tilde{\mathbf{n}}^l\right|^2\right),\end{aligned}$$

where δ is a constant. Therefore, we approximately obtain a minimizer $\tilde{\mathbf{n}}^{l+1}$ of $\mathcal{E}_5^2(\mathbf{n})$ by a cheap arithmetic operation at each grid point:

$$\tilde{\mathbf{n}}^{l+1} = \tilde{\mathbf{n}}^l + \frac{1}{c_4\delta}\left((c_1+\mu_1)\,\tilde{\mathbf{p}}^{l+1} + c_4\nabla\left(\nabla\cdot\tilde{\mathbf{n}}^l - \tilde{g}^{l+1}\right) + \nabla\mu_4\right). \tag{26}$$

Table 1. Computational costs are presented for Fig. 1: number of outer iteration / computational time (sec)

	Fig. 1-(a)	Fig. 1-(b)	Fig. 1-(c)
size	80×80	100×100	300×235
Algorithm in [16]	449/6.57	307/7.75	329/80.64
Method 1	177/2.23	323/6.90	430/79.97
Method 2	187/1.31	445/4.95	383/32.38

More interestingly, the closed form formula (26) is also obtained in a different way. The linear term can be added by an explicit time discretization of the gradient decent method. That is, if we use the gradient descent method for approximately finding a minimizer of (24), we have

$$\frac{\partial \mathbf{n}}{\partial \tau} = (c_1 + \mu_1)\, \tilde{\mathbf{p}}^{l+1} + c_4 \nabla \left(\nabla \cdot \mathbf{n} - \tilde{g}^{l+1} \right) + \nabla \mu_4. \tag{27}$$

Then, explicit Euler scheme gives the same formula as (26) with the time step $\tau = \frac{1}{c_4 \delta}$.

For p and $q = 1$ or 2, there are closed form formulas to find minimizers of $\mathcal{E}_1^2(v)$, $\mathcal{E}_4^1(\mathbf{p})$, and $\mathcal{E}_5^1(g)$ and it takes simply arithmetic operations at each grid point. In the proposed Method 2, we use the GS method for u-subproblem of minimizing $\mathcal{E}_2^2(u)$. Considering a method for low computational cost, one sweep of GS iteration is practically enough for approximately solving the equation. For the $\mathbf{n}$-subproblem $\mathcal{E}_5^2(\mathbf{n})$ (24), we use a simple and cost effective formula (26).

4 Numerical Results

We demonstrate numerical examples using the proposed algorithms in image inpainting and curve reconstruction from unorganized points set. We use the staggered grid system to obtain finite difference discretization of our models; see more details in [16]. The test system is a Intel(R) Core(TM) i7 CPU Q720 1.6GHz with 4GB RAM.

4.1 Image Inpainting

Numerical tests on image inpainting are provided to demonstrate the efficiency of the proposed algorithms. In Fig. 1, we choose the same examples shown in [16]. The inpainting results from Method 1 and Method 2 numerically show that the curvature term works to connect the level curves of image on a large inpainting domain.

In Table 1, the improved computational speed is demonstrated. In order to show efficiency of our algorithms, we use smaller (or same) value of stoping criterion (relative residuals) than [16]. Even though the number of outer iteration is

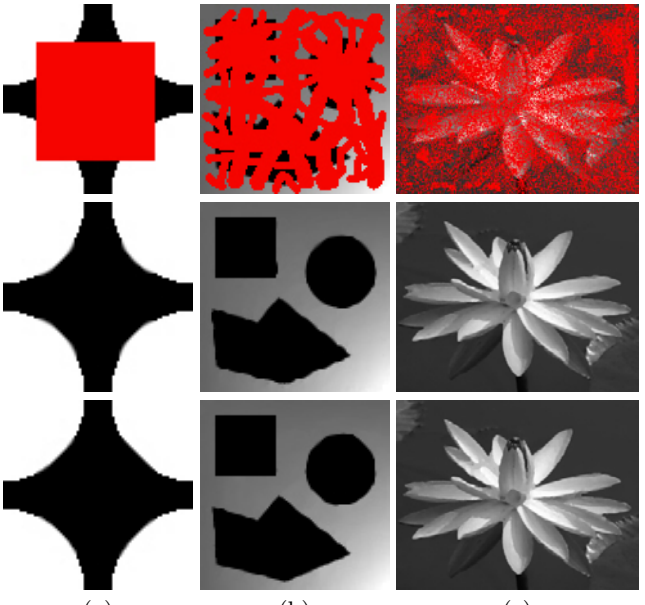


Fig. 1. The red regions in the first row indicate the inpainting domain. The images in the second and third row are image inpainting results from Method 1 and Method 2, respectively. For all results obtained by Method 1, we use $a = 1$ and $\eta = 10^3$. The remaining parameters are $b = 50$, $r_1 = 1$, $r_2 = r_3 = 20$, $r_4 = 10^2$, $r_5 = 1$ in (a), $b = 50$, $r_1 = 1$, $r_2 = 10^2$, $r_3 = 50$, $r_4 = 5 \cdot 10^2$, $r_5 = 10$ in (b), and $b = 30$, $r_1 = 2$, $r_2 = 6 \cdot 10^2$, $r_3 = 10^2$, $r_4 = 10^3$, $r_5 = 10$ in (c). For all results obtained by Method 2, we use $a = 1$, $\eta = 10^3$, and $\delta = 1$. The remaining parameters are $b = 10$, $c_1 = 1$, $c_2 = 5$, $c_3 = 10$, $c_4 = 10^2$ in (a), $b = 50$, $c_1 = 1$, $c_2 = 50$, $c_3 = 10$, and $c_4 = 10^3$ in (b), and $b = 30$, $c_1 = 2$, $c_2 = 4 \cdot 10^2$, $c_3 = 10^2$, and $c_4 = 10^3$ in (c).

larger than the algorithm in [16], the computational time in Method 2 is reduced because we use a very cheap arithmetic operation at each grid point. Method 1 usually may have a similar computational cost to the algorithm in [16] because the number of inner iteration in the frozen coefficient method for solving (12) is empirically less than 5 in an early stage of outer iteration. Moreover, the number of inner iteration tends to be reduced as long as the outer iteration is increased. Since our results are obtained by smaller (or same) relative residual error bound than [16] and they are converged faster than the previous method in Section 2, the proposed algorithms improve the computational cost.

In Fig. 2, we also show the graphs (log scales on xy-axis) of residuals, relative errors in Lagrange multipliers, relative error in u, and energy for Method 2 of the example in Fig. 1-(a); see more details in [16]. The profile of graphs are very similar to the results from Method 1 and [16]. The proposed algorithms are numerically verified that they are practically faster than the previous algorithm in [16].

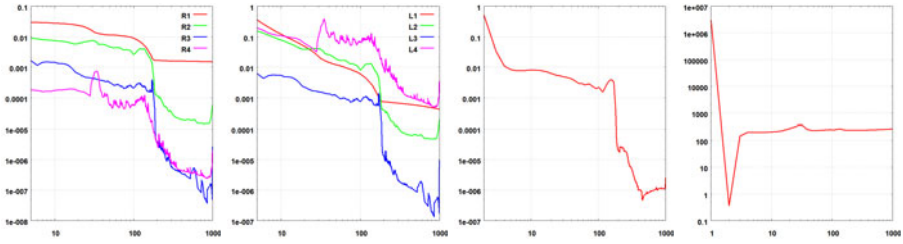


Fig. 2. From the left, the log scale plots of residuals, relative errors in Lagrange multipliers, relative error in u, and energy on y-axis versus iteration on x-axis for Method 2 of the example in Fig. 1-(a). Note that graphs from Method 1 have almost same profiles.

4.2 Curve Reconstruction

To address a reconstruction problem, the following model

$$\int_{\Omega} \left(a\psi + b \left| \nabla \cdot \frac{\nabla u}{|\nabla u|} \right| \right) |\nabla u| + \frac{1}{2} \int_{\Gamma} \eta u((c_1 - u_0)^2 - (c_2 - u_0)^2) \quad \text{s.t.} \quad 0 \le u \le 1$$

is minimized by the proposed methods, where c_1 and c_2 are positive constants and ψ is the unsigned distance function induced from the unorganized points set. As in [18], u_0 is an initial guess obtained with region-growing methods, and the fidelity parameter is a function rather than a constant, which suggests that to what extent the initial guess is faithful. Specifically, we use

$$\eta(x) = \begin{cases} c_\eta & \psi(x) > 5h, \\ 0 & \psi(x) \le 5h, \end{cases}$$

for each point x in the domain Ω, where h is the mesh size and c_η is a constant. For simplicity, we use the domain $\Omega = [-1, 1] \times [-1, 1]$ and we update c_1 and c_2 for every 100 iterations in our implementation as [11]:

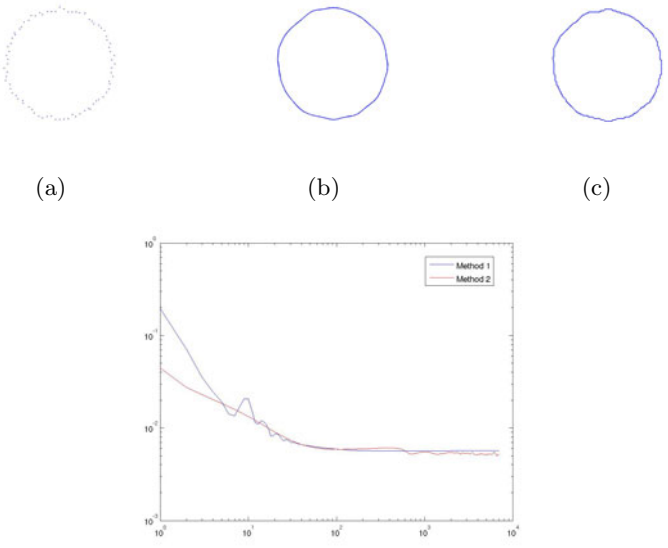
$$c_1 = \frac{1}{\mathcal{A}(\mathcal{R})} \int_{\mathcal{R}} u \quad \text{and} \quad c_2 = \frac{1}{\mathcal{A}(\Omega \setminus \mathcal{R})} \int_{\Omega \setminus \mathcal{R}} u,$$

where $\mathcal{R} \equiv \{x \in \Omega : u(x) \ge 0.5\}$ and $\mathcal{A}(\cdot)$ measures the area of a set.

To impose the constraint on u, a projection operator in [11] is carried on a new variable v in the augmented Lagrangian formulations in Sections 2 and 3. For example, in Method 1, the v-subproblem is solved as follows:

$$\begin{cases} \tilde{v} = \arg\min_{\tilde{v}} \tilde{\mathcal{E}}_1(v) = \arg\min_{\tilde{v}} \int_{\Omega} \frac{\eta}{2} \tilde{v} f + \int_{\Omega} \frac{r_3}{2} \left(\tilde{v} - \tilde{u}^l \right)^2 + \lambda_3^k \tilde{v}, \\ v = \max\{\min\{\tilde{v}, 0\}, 1\}. \end{cases}$$

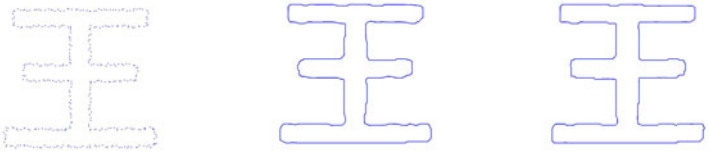
Since u converges to v, the constraint is therefore imposed on u correspondingly. The same projection operation is used in Method 2 as well. In all experiments, it is observed that u converges to a function between 0 and 1. Figure 3 presents a reconstruction example.



(a) (b) (c)

(d)

Fig. 3. (a) Noisy points set sampled from a circle; (b) The result produced by Method 1; (c) The result produced by Method 2; (d) Error (28) vs. iteration times (loglog). In this example, we use $a = 1$, $b = 10^4$, $t = 2 \cdot 10^4$, $r_1 = 1$, $r_2 = 0.1$, $r_3 = 2 \cdot 10^2$, $r_4 = 2$, and $r_5 = 10^2$ in Method 1. The parameters for Method 2 are $a = 0.5$, $b = 10^4$, $t = 2 \cdot 10^4$, $c_1 = 1$, $c_2 = 0.1$, $c_3 = 1.5 \cdot 10^2$, $c_4 = 30$, and $\delta = 1$.



(a) (b) (c)

Fig. 4. Results from different b in Method 2: (a) Noisy point set; (b) Reconstructed curve with $b = 1$; (c)Reconstructed curve with $b = 10$. The remaining parameters are selected as $a = 1$, $t = 2 \cdot 10^4$, $c_1 = 1$, $c_2 = 0.1$, $c_3 = 2 \cdot 10^2$, $c_4 = 2$, and $\delta = 1$.

Fig. 3-(a) shows the points set with 5% noise sampled from a unit circle. From the noisy points set, Fig. 3-(b) illustrates the reconstructed circle produced by Method 1 and Fig. 3-(c) is the result with Method 2. It can be seen that Method 1 yields smoother result because it solves (18) completely. Fig. 3(d) gives the

error against iteration times. Here, the error is the measurement of comparison between the reconstructed curves and the exact unit circle as follows:

$$E_u = \frac{|\mathbf{1}_{exact} - \mathbf{1}_u|_{L^1}}{|\mathbf{1}_{exact}|_{L^1}}, \tag{28}$$

where $\mathbf{1}_u$ and $\mathbf{1}_{exact}$ are the indicator functions, which take value 1 or 0 for each point in the domain. $\mathbf{1}_{exact}$ is an indicator function of the circle centered at origin and with radius 0.3

$$\mathbf{1}_{exact}(x) = \begin{cases} 1 & \|x\| \leq 0.3, \\ 0 & \|x\| > 0.3, \end{cases}$$

where $|x|$ is the Euclidean length of the point x and

$$\mathbf{1}_u(x) = \begin{cases} 1 & u(x) > 0.5, \\ 0 & u(x) \leq 0.5. \end{cases}$$

Although the convergent rates of Methods 1 and 2 are almost similar, Method 2 is much faster than Method 1 for each iteration.

Fig. 4 gives another reconstruction example by Method 2. Fig. 4-(a) shows noisy points set sampled from a chinese character. Figs. 4-(b) and 2-(c) illustrate reconstructed curves with different parameters. We numerically observe that a better result is obtained by increasing parameter b.

5 Conclusion

We proposed two algorithms to efficiently solve the p-elastica model in image inpainting and curve reconstruction from unorganized point set. Inspired by the recent work [16], we used augmented Lagrangian method and extend the algorithm in [16]. The first algorithm eliminates an inner iterative steps in [16] and the second algorithm replaces FFT into a very cheap arithmetic operation. From the numerical results, the efficiency of the algorithms are demonstrated. In the future, we would like to extend the model in subsection 4.2 into 3D to reconstruct a surface which minimizes its mean curvature. Moreover, the Euler's elastica model on the surface is a possible extension of using the proposed Method 2.

References

1. Ambrosio, L., Masnou, S.: A direct variational approach to a problem arising in image reconstruction. Interfaces Free Bound 5(1), 63–81 (2003)
2. Ardentov, A., Sachkov, Y.L.: Solution to Euler's elastica problem. Automation and Remote Control 70, 633–643 (2009)
3. Bae, E., Shi, J., Tai, X.C.: Graph cuts for curvature based image. IEEE Trans. Image Process (to appear)

4. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Trans. Image Processing 10(8), 1200–1211 (2001)
5. Ballester, C., Caselles, V., Verdera, J.: Disocclusion by joint interpolation of vector fields and gray levels. Multiscale Model. Simul. 2(1), 80–123 (2003)
6. Bellettini, G., Dal Maso, G., Paolini, M.: Semicontinuity and relaxation properties of a curvature depending functional in 2d. Ann. Scuola Norm. Sup. Pisa Cl. Sci (4) 20(2), 247–297 (1993)
7. Bruckstein, A., Netravali, A., Richardson, T.: Epi-convergence of discrete elastica. Appl. Anal. 79, 137–171 (2001)
8. Chan, T.F., Kang, S.H., Shen, J.: Euler's elastica and curvature based inpaintings. SIAM J. Appl. Math. 63(2), 564–594 (2002)
9. Chan, T.F., Shen, J.: Nontexture inpainting by curvature driven diffusion (CDD). J. Visul Comm. Image Rep. 12, 436–449 (2001)
10. El-Zehiry, N., Grady, L.: Fast global optimization of curvature. In: IEEE Conference on CVPR, pp. 3257–3264 (2010)
11. Goldstein, T., Bresson, X., Osher, S.: Geometric applications of the split bregman method: Segmentation and surface reconstruction. Journal of Scientific Computing 45, 272–293 (2010)
12. Lai, R., Chan, T.F.: A framework for intrinsic image processing on surfaces. Tech. rep., UCLA CAM Report 10-25 (2010)
13. Masnou, S., Morel, J.M.: Level lines based disocclusion. In: Proc. IEEE Int. Conf. on Image Processing, Chicago, IL, pp. 259–263 (1998)
14. Nitzberg, M., Mumford, D., Shiota, T.: Filtering, Segmentation and Depth, vol. 662. Springer, Berlin (1993)
15. Schoenemann, T., Kahl, F., Cremers, D.: Curvature regularity for region-based image segmentation and inpainting: A linear programming relaxation. In: IEEE Int. Conf. Comp, Vision, Kyoto, Japan (2009)
16. Tai, X.C., Hahn, J., Chung, G.J.: A fast algorithm for Euler's elastica model using augmented Lagrangian method. SIAM J. Img. Sci. 4, 313–344
17. Tai, X.-C., Wu, C.: Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 502–513. Springer, Heidelberg (2009)
18. Wan, M., Wang, Y., Wang, D.: Variational surface reconstruction based on delaunay triangulation and graph cut. International Journal for Numerical Methods in Engineering 85, 206–229 (2011)
19. Wu, C., Tai, X.C.: Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models. SIAM J. Imaging Sci. 3(3), 300–339 (2010)
20. Yun, S., Woo, H.: Linearized proximal alternating minimization algorithm for motion deblurring by nonlocal regularization. In: Pattern Recognition (to appear in, 2011)