Augmented Lagrangian method for generalized TV-Stokes model *

Jooyoung Hahn[†], Chunlin Wu[‡], and Xue-Cheng Tai[§]

[†,‡,§]Division of Mathematical Sciences, School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore
[§]Mathematics Institute, University of Bergen, Norway

Abstract

In this paper, we propose a general form of TV-Stokes models and provide an efficient and fast numerical algorithm based on the augmented Lagrangian method. The proposed model and numerical algorithm can be used for a number of applications such as image inpainting, image decomposition, surface reconstruction from sparse gradient, direction denoising, and image denoising. Comparing with properties of different norms in regularity term and fidelity term, various results are investigated in applications. We numerically show that the proposed model recovers jump discontinuities of a data and discontinuities of the data gradient while reducing stair-case effect.

Keyword: TV-Stokes model, Augmented Lagrangian method, Image inpainting, Image decomposition, Surface reconstruction from sparse gradient, Direction denoising, Image denoising

1 Introduction

In image denoising and image inpainting, one of main goals is to recover local structures in an original image. Even though there are no certain hierarchical local structures which decide the visual quality of recovered image, human vision easily tends to detect edges, ridges or valleys, and smooth regions [1]. We regard these features as jump discontinuities of a data which are considered as edges in an image, discontinuities of the data gradient which are possibly interpreted as sharp ridges or sharp valleys, and smooth changes of the data where its gradient field is smooth. In this paper, we explain that the proposed general form of TV-Stokes model reconstructs these local features and provide an efficient and fast numerical algorithm to solve the proposed model via the augmented Lagrangian method. We also show that the proposed model and numerical algorithm can be used for a number of applications.

The Rudin-Osher-Fatemi (ROF) model [2] has been a framework in variational image denoising. We also call this model as the TV-L^2 model. Since the TV regularization reduces oscillatory features while allowing jump discontinuities of an image, the ROF model is suitable to denoise piecewise constant images and has been used for many applications [3–5]. Some drawbacks in the ROF model were noticed such as reduction of contrast and stair-case effect [6–10]. The authors [9] showed that the TV-L^2 model never recover the same contrast in case of a simple disk shape no matter how

*The research is supported by IDM project NRF2007IDMIDM002-010 and MOE (Ministry of Education) Tier II project T207N2202.

[†]jyhahn@ntu.edu.sg
[‡]CLWU@ntu.edu.sg
[§]xctai@ntu.edu.sg

large the fidelity constant is used. In order to preserve the contrast of an image, the L^1 fidelity was proposed by Alliney for a discrete one dimensional data [11]. In [12], Nikolova pointed out that the TV-L^1 model yields the minimizer which has exactly same values as a given image at some pixels and the analysis was applied to more general fidelity terms. In [10], the authors addressed the continuous analogues to the discrete version of the TV-L^1 model [11–13]. In order to reduce stair-case effect in image denoising, the energy functional involving higher order derivatives was introduced in [7, 14–19]. The Euler-Lagrange equation is a nonlinear fourth-order partial differential equation (PDE) and the minimizer allows discontinuities of an image and its gradients. There are other fourth-order models from a PDE-based approach [17, 20] and hybrid models with a filter-based approach [21, 22]. In [23], stair-case effect in the Perona-Malik model [24] is also removed by a modification of the neighborhood filter which is a linear regression correction.

Recently, there have been many researches for fast and accurate algorithms to obtain a minimizer of energy functional with TV regularization and L^p fidelity ($p = 1$ or 2) such as the dual methods [5, 25–30], Bregman iterative algorithm [31], split Bregman iteration [32, 33], alternating minimization algorithm [34–36], Douglas-Rachford splitting [37, 38], and the references therein. The augmented Lagrangian method [39–41] is used for more general models, such as the vectorial TV model, the higher order model, and the TV-L^p ($p \geq 1$) model. In [42], the Uzawa block relaxation methods to the corresponding augmented Lagrangian functional of a weighted TV regularization model were addressed to solve a geometric filtering of the image components. Many fast algorithms and general framework for first order primal-dual algorithms for TV regularization are well explored in [29, 43, 44]. The dual formulation of the TV-Stokes model [45] is also proposed for image denoising [46].

In image inpainting, the Masnou and Morel (MM) model [47] and Bertalmio-Sapiro-Caselles-Ballester (BSCB) model [48], indicate that it is difficult to obtain some pleasant inpainted results from second order PDEs. The authors in [47] proposed an energy functional which minimizes the length of the level lines and the angle total variation. The Euler-Lagrange equation is a nonlinear fourth order PDE and the level curves are linearly connected from the outside of the inpainting domains. The BSCB model was designed by a transportation phenomenon written as a nonlinear third order PDE. The property of total variation inpainting, which yields the Euler-Lagrangian equation as a nonlinear second order PDE, is mathematically studied and compared with the harmonic inpainting in [49]. The authors [50] proposed an approach to estimate the direction of extended level lines and the other and to reconstruct the image whose gradient direction fits to the estimated one. In [51], an analogy to fluid dynamics is used. If the stream function is an image I, the fluid velocity in the incompressible Newtonian fluids is $\nabla^{\perp} I$. The inpainting process can be considered as transporting the vorticity into the inpainting regions with the anisotropic diffusion and then reconstructing the image via the Poisson equation. The generalization of the MM model was introduced via Euler's elastica curve in [52]. The Euler-Lagrange equation includes two crucial processes in image inpainting, transportation [48] and diffusion [53], which yield excellent results in many challenging inpainting regions such as a smooth connection of level lines over a large inpainting domain. The PDE-based approach to overcome the long connectivity is also introduced in [54, 55].

The Lysaker-Osher-Tai (LOT) model [56] consists of two steps in image denoising. The unit gradient vector field is regularized via the TV norm in the first step and then a denoised image is reconstructed by fitting the image gradient into the regularized vector. The method outperforms the ROF model [2] and the Lysaker-Lundervold-Tai (LLT) model [16]. Recently, the authors [57] improved the LOT model by regularizing angle instead of vectors and using an edge indicator as an extra weight. In image inpainting, the two-step method in the LOT model is improved with the divergence free constraint [58]. The authors proposed the incompressibility condition inspired by an analogy between incompressible Navier-Stokes equations and image inpainting in [51]. The incompressibility condition enforces a propagation of tangential vectors over large inpainting regions. The compatible condition to the normal vector field is called as the integrability condition and it has been extensively used in surface reconstruction from a dense normal field in computer vision; see [59–62] and references therein. In [45], the same incompressibility condition is used in image denoising and the authors numerically showed that the TV-Stokes method is good for preserving edges while reducing stair-case effect. Some mathematical properties of TV-Stokes method with the anisotropic TV-norm were recently studied in [63]. The authors point out that the energy functional in the reconstruction step in [56] does not make sense in the bounded variation (BV) space if the regularized vector field is not smooth. They proposed a mathematically sound energy functional in surface reconstruction and the existence and the uniqueness of the minimizer were proved. We would like to mention that two separate procedures, estimating derivatives of a surface and reconstructing a surface whose derivatives fit to the estimated ones, are already well-known methods in computer vision such as shape from shading [64], mesh regularization [65], computer graphics [66–69], photometric stereo [61], and single view modeling [70, 71].

Contemporary TV-Stokes models consist of two steps: the first step is to regularize the tangential vector field to the level curves of the image and the second step is to reconstruct the image whose gradient fits in the regularized normal vector obtained in the first step. In this paper, we generalize the first step in [45, 58] and the second step in the modified TV-Stokes model [63]. The generalized form uses TV or H^1 regularization, the L^p fidelity ($p \geq 1$), and the arbitrary integration domain in the fidelity term. Note that the integration domain can be extended to include open or closed curves. This extension makes it possible to use the proposed model to find a surface from sparse gradient, which is a well-known problem in computer vision [61, 64, 70] and computer graphics [67–69]. One of main focus in this paper is to provide an efficient and fast numerical algorithm in order to solve the generalized models based on the augmented Lagrangian method. We also explain a property of the generalized TV-Stokes model which is to decompose an image into jump discontinuities of a data and discontinuities of the data gradient or smooth regions. The generalized model can be used a number of applications such as image inpainting, image decomposition, surface reconstruction from sparse gradient, direction denoising, and image denoising. We investigate the various effects using different norms via many examples in applications and numerically show that the proposed model recovers jump discontinuities of a data and discontinuities of the data gradient while reducing stair-case effect.

The paper is organized as follows. In Section 2, we propose the general form of TV-Stokes models and explain some basic properties of the proposed model. In Section 3, the augmented Lagrangian method is used to solve the proposed general model. We present numerical examples in Section 4. The paper concludes in Section 5.

2 Generalized TV-Stokes model

In this section, we propose a generalized TV-Stokes model in Section 2.1 and show relation to previous models in Section 2.2. Some basic properties of the proposed functional are explained in Section 2.3.

2.1 Proposed model

First of all, we would like to introduce a very simple model which consists of two steps in image denoising with Gaussian white noise. The two procedures are regularization of image derivative and then reconstruction of denoised image from the regularized image derivative. This separate procedure is based on a very naive concept - "more accurate image gradient makes better reconstruction". In image denoising, the Perona-Malik (PM) model [24, 72] uses estimation of image gradient as derivative of the convolution of image with the two-dimensional Gaussian kernel. The coherence-enhancing diffusion [73] uses more accurate derivative estimation from the structure tensor. The derivative estimation in [73] makes it possible to denoise flow-like images which are difficult to be denoised by the PM model.

We denote Ω as a rectangular domain. Let us consider an image $I : \Omega \subset \mathbf{R}^2 \to [0, 1]$. The normal and tangential vectors of the level curves of the image are given by

$$\mathbf{n} = \nabla I = (\partial_1 I, \partial_2 I)^{\mathrm{T}} \quad \text{and} \quad \mathbf{t} = \nabla^{\perp} I = (\partial_2 I, -\partial_1 I)^{\mathrm{T}}.$$

These vector fields satisfy the irrotational condition and the incompressible condition almost everywhere in Ω:

$$\nabla \times \mathbf{n} = 0 \quad \text{and} \quad \nabla \cdot \mathbf{t} = 0. \tag{2.1}$$

A general type of vector field does not need to satisfy the condition (2.1). However, if we regularize a vector field $\mathbf{n}$ (or $\mathbf{t}$) and we use $\mathbf{n}$ (or $\mathbf{t}$) to find a function g such that $\nabla g = \mathbf{n}$ (or $\nabla^{\perp} g = \mathbf{t}$), then the condition $\nabla \times \mathbf{n} = 0$ (or $\nabla \cdot \mathbf{t} = 0$) is necessary. Since the conditions are violated on edges and ridges or valleys in an image I, it is difficult to impose the conditions pointwisely.

Now, we introduce a very simple TV-Stokes model which consists of two steps in image denoising. From a noisy image I^*, we have a noisy vector field $\mathbf{t}^* = \nabla^{\perp} I^*$. In the first step, we obtain a regularized vector field $\mathbf{t} = (t_1, t_2)^{\mathrm{T}}$ via an energy functional minimization:

$$\min_{\nabla \cdot \mathbf{t} = 0} \int_{\Omega} |\nabla \mathbf{t}| + \eta \int_{\Omega} |\mathbf{t} - \mathbf{t}^*|, \tag{2.2}$$

where

$$|\nabla \mathbf{t}|^2 = (\partial_1 t_1)^2 + (\partial_2 t_1)^2 + (\partial_1 t_2)^2 + (\partial_2 t_2)^2.$$

The TV regularization preserves discontinuities in regularized derivative and it will help to preserve sharp ridges or valleys in image denoising. The divergence free constraint is reasonable because we regularize vector field $\mathbf{t}$ from $\mathbf{t}^* = \nabla^{\perp} I^*$. In the second step, we integrate the regularized normal vector field $\mathbf{n} = \mathbf{t}^{\perp}$ obtained by (2.2) to reconstruct a denoised image via an energy functional minimization:

$$\min_{I} \int_{\Omega} |\nabla I - \mathbf{n}| + \frac{\xi}{2} \int_{\Omega} |I - I^*|^2. \tag{2.3}$$

A denoised image is obtained by minimizing a difference between image gradient and the regularized normal vector field from the first step. Note that the second step is exactly the same as the TV-L^2 model [2] when $\mathbf{n} = 0$. We expect that the TV type regularization in (2.3) preserves jump discontinuities in a denoised image. We shall carefully investigate two models (2.2) and (2.3) through many examples.

The reason we use the L^1 fidelity in (2.2) is a compatibility between two norms: the fidelity term in (2.2) and the regularization term in (2.3). Since $\mathbf{t}$ is the regularized vector field of $\mathbf{t}^* = \nabla^{\perp} I^*$, the fidelity term in (2.2) measures the difference of image gradient. Moreover, the regularization term in (2.3) also has the same meaning. That is, it would be desirable to use same norms for these quantities. Since the TV regularization should be used in the second step in order to reconstruct jump discontinuities in a denoised image, the compatible and reasonable choice of fidelity in the first step is the L^1 fidelity. In Section 4, we numerically show the advantage of L^1 fidelity in (2.2) in some applications.

In this paper, we would like to extend the use of TV-Stokes model (2.2) and (2.3) into many other applications such as image inpainting, image decomposition, and surface reconstruction from sparse gradient. Moreover, we want to numerically investigate the various effects of different regularizations and fidelities in (2.2) and (2.3). For this purpose, we propose a generalized form of TV-Stokes model (2.2) and (2.3). The generalized model also consists of two steps: regularization of data derivative in the first step and then reconstruction of data in the second step based on the regularized derivative. The model can be considered as the generalized form of the first step in [45, 58] and the second step in [63]. The general form uses TV or H^1 regularization, the L^p fidelity ($p \geq 1$), and the arbitrary integration domain in the fidelity term.

We consider a data $I^* : \Gamma \subset \Omega \to [0, 1]$ and its tangential vector field $\mathbf{t}^* = \nabla^\perp I^*$ on Γ. In the first step of the general model, we obtain a regularized tangential vector field $\mathbf{t}$ from a given vector field $\mathbf{t}^*$ on $\Gamma \subset \Omega$ via the following energy functional minimization:

$$\min_{\mathbf{t}} \int_\Omega |\nabla \mathbf{t}|^q + \frac{\eta}{p} \int_\Gamma |\mathbf{t} - \mathbf{t}^*|^p, \quad \text{subject to } F(\mathbf{t}) = 0, \tag{2.4}$$

where $p \geq 1$, $q = 1$ or 2, and $\eta > 0$. The function F represents the extra constraint of the vector field such as the incompressibility condition.

The choice of $\mathbf{t}^*$ and Γ depends on applications. In image denoising and image decomposition, I^* and $\mathbf{t}^* = \nabla^\perp I^*$ are noisy data and $\Gamma = \Omega$ is the domain of image I^*. In image inpainting, we have inpainting regions $\mathcal{R}$ and then $\Gamma = \Omega \setminus \mathcal{R}$ and $\mathbf{t}^* = \nabla^\perp I^*$ are used. I^* on the boundary of $\mathcal{R}$ has the information to be inpainted on $\mathcal{R}$. Since we have no restriction to choose $\Gamma \subset \Omega$, the integration domain Γ of the fidelity term in (2.4) can have not only regions but also closed or open curves. In case of surface reconstruction from sparse gradient, Γ is a collection of curves to indicate the location of the given gradient vectors $\mathbf{n}^*$ and $(\mathbf{t}^*)^\perp = \mathbf{n}^*$; see Figure 4.5. In direction denoising, $\Gamma = \Omega$ is the domain of noisy vector data $\mathbf{t}^*$ and $F \equiv 0$. In these applications, we consider four different cases to obtain a regularized vector field and the following abbreviations are used to denote different minimization:

- $F = \nabla\cdot$ and $q = 1:$ TV-Stokes minimization with L^p fidelity (TVS-L^p).
- $F = \nabla\cdot$ and $q = 2:$ H^1-Stokes minimization with L^p fidelity (H^1S-L^p).
- $F \equiv 0$ and $q = 1:$ vectorial TV minimization with L^p fidelity (VTV-L^p).
- $F \equiv 0$ and $q = 2:$ vectorial H^1 minimization with L^p fidelity (VH^1-L^p).

In the second step, we already have the regularized normal vector field $\mathbf{n} = \mathbf{t}^\perp$ from the first step. We use another general model to reconstruct a data via the following energy functional minimization:

$$\min_I \int_\Omega |\nabla I - \mathbf{n}|^q + \frac{\xi}{p} \int_\Gamma |I - I^*|^p, \tag{2.5}$$

where $p \geq 1$, $q = 1$ or 2, $\xi > 0$. Note that p and q can be different from (2.4). We also use the following abbreviations to indicate different models:

- $q = 1:$ TV$\mathbf{n}$ minimization with L^p fidelity (TV$\mathbf{n}$-L^p).
- $q = 2:$ $H^1\mathbf{n}$ minimization with L^p fidelity ($H^1\mathbf{n}$-L^p).

If we need to indicate a model which consists of H^1-Stokes minimization with the L^2 fidelity for regularizing a vector field and TV$\mathbf{n}$ minimization with the L^1 fidelity for reconstructing an image, we simply use the abbreviation as H^1S-L^2+TV$\mathbf{n}$-L^1. In case that there is no prior data I^* in (2.5), we use $\xi = 0$ and fix a data value as zero at a point in Ω. We denote this model as TV$\mathbf{n}$ ($q = 1$) and $H^1\mathbf{n}$ ($q = 2$) in (2.5). These models are used in image decomposition and surface reconstruction from sparse gradient.

In the rest of this paper, our main focus is to provide a fast and efficient algorithm to minimize (2.4) and (2.5) based on the augmented Lagrangian method in Section 3. We also investigate various effects of different norms via many examples in Section 4. As the TV regularization preserves discontinuities of data and the H^1 regularization enforces the continuity of data, we will observe similar phenomena when a tangential vector field is regularized in (2.4) and a data is reconstructed in (2.5). Moreover, we will see the effect of L^1 or L^2 fidelity in (2.4) via different applications.

2.2 Relation to previous models

The basic TV-Stokes model (2.2) and (2.3) which we generalize in this paper are related with previous works. The authors [56] used a regularization of normal vector field with the unit length constraint and a denoised image is recovered by fitting the image gradient into the regularized normal vector field:

$$\min_{|\mathbf{n}|=1} \int_\Omega |\nabla \mathbf{n}| + \frac{\eta}{2} \int_\Omega |\mathbf{n} - \mathbf{n}^*|^2, \tag{2.6}$$

$$\min_I \int_\Omega (|\nabla I| - \nabla I \cdot \mathbf{n}) + \frac{\xi}{2} \int_\Omega |I - I^*|^2, \tag{2.7}$$

where I^* is a noisy image and $\mathbf{n}^* = \nabla I^*/|\nabla I^*|$. In [50], the regularizer in (2.7) is used in image inpainting. The continuation of image gradient is considered and an image is simultaneously recovered by an energy functional minimization:

$$\min_{\mathbf{n},I} \left(\int_\mathcal{R} |\nabla \cdot \mathbf{n}|^p \left(c_1 + c_2 |\nabla k * I|\right) + \zeta \int_\mathcal{R} \left(|\nabla I| - \nabla I \cdot \mathbf{n}\right) \right), \tag{2.8}$$

where k denotes a regularizing kernel, $\mathcal{R}$ is an inpainting domain, and the detail admissible sets and other variables are explained in [50]. In image inpainting, the LOT model in [56] is improved with the divergence free constraint. The authors [58] proposed a minimization model:

$$\min_{\nabla\cdot\mathbf{t}=0} \int_\mathcal{R} |\nabla \mathbf{t}| \, d\mathbf{p}, \quad \text{with} \quad \mathbf{t} = \mathbf{t}^* \text{ on } \partial\mathcal{R}. \tag{2.9}$$

$$\min_I \int_\mathcal{R} \left(|\nabla I| - \nabla I \cdot \frac{\mathbf{n}}{|\mathbf{n}|} \right) d\mathbf{p}, \quad \text{with} \quad I = I^* \text{ on } \partial\mathcal{R}. \tag{2.10}$$

Instead of regularizing normal vector field with unit length constraint in (2.6), the incompressible condition of tangential vector field enforces the propagation of vector field into a large inpainting domain. In [45], the same method is used in image denoising. In [63], the authors point out that the energy functional in (2.7) does not make sense in the BV space if the regularized vector field $\mathbf{n}$ is not smooth. They used a different reconstruction step in image denoising:

$$\min_{\nabla\cdot\mathbf{t}=0} \int_\Omega \sum_{l=1}^{2} |\nabla t_l| + \frac{\eta}{2} \int_\Omega |\mathbf{t} - \mathbf{t}^*|^2, \tag{2.11}$$

$$\min_I \int_\Omega |\nabla I - \mathbf{n}| + \frac{\xi}{2} \int_\Omega |I - I^*|^2, \tag{2.12}$$

where $\mathbf{t} = (t_1, t_2)$. The basic models (2.2) and (2.3) are very similar to the above model except the isotropic TV regularization and the L^1 fidelity in the first step.

The variational model to recover jump discontinuities in an image and discontinuities of the image gradient in image denoising was introduced in [14]. The authors regarded an image as $I = u_1 + u_2$, where $u_1 \in \mathrm{BV}(\Omega)$ and $u_2 \in H^1$ with $\nabla u_2 \in \mathrm{BV}(\Omega, \mathbf{R}^2)$ and then proposed to denoise an image from a noisy data I^* via the minimization:

$$\begin{aligned} &\min_{u_1,u_2} \int_\Omega |\nabla u_1| + \alpha \int_\Omega |\nabla^2 u_2| + \lambda \int_\Omega (u_1 + u_2 - I^*)^2 \\ &= \min_{I,h} \int_\Omega |\nabla I - \nabla h| + \alpha \int_\Omega |\nabla^2 h| + \lambda \int_\Omega (I - I^*)^2 , \end{aligned} \tag{2.13}$$

where $|\nabla^2 h|^2 = |\nabla(\partial_1 h)|^2 + |\nabla(\partial_2 h)|^2$ and α and λ are positive constants. Introducing $\mathbf{n} = \nabla h$ and using $|\nabla \mathbf{n}| = |\nabla \mathbf{t}| = |\nabla^2 h|$, the minimization (2.13) can be written as follows:

$$
\begin{aligned}
&\min_{\mathbf{n}=\nabla h} \int_\Omega |\nabla I - \mathbf{n}| + \alpha \int_\Omega |\nabla \mathbf{n}| + \lambda \int_\Omega (I - I^*)^2 , \\
&= \min_{\nabla\times\mathbf{n}=0} \int_\Omega |\nabla I - \mathbf{n}| + \alpha \int_\Omega |\nabla \mathbf{n}| + \lambda \int_\Omega (I - I^*)^2 , \\
&= \min_{\nabla\cdot\mathbf{t}=0} \int_\Omega |\nabla I - \mathbf{n}| + \alpha \int_\Omega |\nabla \mathbf{t}| + \lambda \int_\Omega (I - I^*)^2 ,
\end{aligned}
$$

where $\mathbf{n} = \mathbf{t}^\perp$ and $\mathbf{t}^* = \nabla^\perp I^*$. Inspired by the last minimization, we introduced the first step (2.2) in order to regularize the tangential vector field under the TV norm and the divergence free constraint. Since the tangential vector field is regularized independently, a fidelity term is necessary in the first step. After a regularized tangential vector field is obtained, an image is recovered by the second step in (2.3) which is the same method as (2.12).

In this paper, we generalize the basic TV-Stokes model (2.2) and (2.3). We shall show that the proposed general model can be used for many applications and investigate various effects of different regularizations and fidelities via many examples in Section 4.

2.3 Properties of the proposed model

Before we close Section 2, we would like to explain some properties of proposed general model. As many higher models [7,14–16,18,19] achieved to obtain edges, sharp ridges or valleys, and smooth regions, the proposed TVS-L^{p_1}+TV$\mathbf{n}$-L^{p_2} model ($p_1, p_2 \geq 1$) also preserves these local structures very well.

First of all, the TV regularization in the TVS-L^{p_1} model is the same as the total variation of image gradient shown in many higher order models. Since the TV regularization of image gradient allows discontinuities in image gradient, the regularized vector field in the TVS-L^{p_1} model also preserves such discontinuities. Moreover, as the authors in [56, 74] mentioned, the first step in the TVS-L^{p_1}+TV$\mathbf{n}$-L^{p_2} model minimizes the same regularity in the LLT model [16]. The regularized vector field $\mathbf{t}$ in the TVS-L^{p_1} model is observed to have the same advantage in the LLT model, which is to preserve discontinuities of the image gradient and to reduce stair-case effect on smooth regions.

To complete two-step method of TVS-L^{p_1}+TV$\mathbf{n}$-L^{p_2} model, the TV$\mathbf{n}$-L^{p_2} model should be solved in the second step. We expect that jump discontinuities of the image are going to be recovered in this step. Since we penalize the divergence free condition in the first step, there exists an image g such that

$$
\nabla g = \mathbf{n}. \tag{2.14}
$$

Then, the TV$\mathbf{n}$-L^{p_2} model is written as

$$
\begin{aligned}
&\min_I \int_\Omega |\nabla I - \mathbf{n}| + \frac{\xi}{2} \int_\Omega |I - I^*|^{p_2} \\
&= \min_I \int_\Omega |\nabla I - \nabla g| + \frac{\xi}{2} \int_\Omega |(I - g) - (I^* - g)|^{p_2} \\
&= \min_f \int_\Omega |\nabla f| + \frac{\xi}{2} \int_\Omega |f - f^*|^{p_2},
\end{aligned} \tag{2.15}
$$

where $f^* = I^* - g$. From the minimizer f, the image is reconstructed by

$$
I = f + g. \tag{2.16}
$$

Therefore, the image f preserves discontinuities of an image as the ROF model ($p_2 = 2$) does and the image g preserves discontinuities of the image gradient and smooth regions of the image as the

LLT model does. Note that the function g in (2.14) is numerically obtained by the TV**n** model with $\xi = 0$ in (2.5).

As Chambolle and Lions [14] proposed the decomposition of an image $u = u_1 + u_2$ in (2.13), we also have such a decomposition property in the proposed general model. More precisely, the TVS-L^1+TV**n**-L^2 model decomposes an image into jump discontinuities of an image and discontinuities of the image gradient or smooth regions; see more details in Section 4.2. Note that the difference between (2.13) and the TVS-L^1+TV**n**-L^2 model is the use of L^1 fidelity. We also emphasize the advantage of the divergence free constraint in image inpainting in Section 4.1.

3 Augmented Lagrangian Method

In this section, we explain the detail algorithms to solve the proposed general model based on the augmented Lagrangian method. In subsection 3.1, we introduce some notations. The proposed algorithms of the first step (2.4) and the second step (2.5) are shown in subsection 3.2 and subsection 3.3, respectively. We mainly show how to incorporate the augmented Lagrangian method with the staggered grid system and how to solve coupled PDEs caused by the incompressibility condition.

3.1 Notation

Let $\Omega = [1, N_1] \times [1, N_2]$ be a set of $N_1 N_2$ points in $\mathbf{R}^2$. For the simplicity, we denote four inner product vector spaces:

$$
\begin{aligned}
X &= \mathbf{R}^{N_1 N_2}, \quad \mathbf{X} = \underbrace{X \times \cdots \times X}_{k}, \\
Y &= X \times X, \quad \mathbf{Y} = \underbrace{Y \times \cdots \times Y}_{k}.
\end{aligned}
$$

As the coordinate (i,j) denotes,

$$
\begin{aligned}
u \in X, &\quad u(i,j) \in \mathbf{R}, \\
p \in Y, &\quad p(i,j) = (p_1(i,j), p_2(i,j)) \in \mathbf{R}^2, \\
\mathbf{u} \in \mathbf{X}, &\quad \mathbf{u}(i,j) = (u_1(i,j), \ldots, u_k(i,j)) \in \mathbf{R}^k, \\
\mathbf{p} \in \mathbf{Y}, &\quad \mathbf{p}(i,j) = (p^1(i,j), \ldots, p^k(i,j)) \in \underbrace{\mathbf{R}^2 \times \cdots \times \mathbf{R}^2}_{k},
\end{aligned}
$$

we equip the standard Euclidean inner products as follows:

$$
\begin{aligned}
(u,v)_X &\equiv \sum_{i,j} u(i,j)v(i,j), & (\mathbf{u},\mathbf{v})_{\mathbf{X}} &\equiv \sum_{l=1}^{k} (u_l, v_l)_X, \\
(p,q)_Y &\equiv (p_1,q_1)_X + (p_2,q_2)_X, & (\mathbf{p},\mathbf{q})_{\mathbf{Y}} &\equiv \sum_{l=1}^{k} (p^l, q^l)_Y.
\end{aligned}
$$

Note that the induced norms $||\cdot||_V$ are the ℓ_2-norm on vector spaces $V = X$, Y, $\mathbf{X}$, and $\mathbf{Y}$. We denote the absolute value as the Euclidean distance.

The discrete backward and forward differential operators for $u \in X$ are defined with the periodic boundary condition:

$$\partial_1^- u(i,j) \equiv \begin{cases} u(i,j) - u(i-1,j), & 1 < i \le N_1, \\ u(1,j) - u(N_1,j), & i = 1, \end{cases}$$

$$\partial_2^- u(i,j) \equiv \begin{cases} u(i,j) - u(i,j-1), & 1 < j \le N_2, \\ u(i,1) - u(i,N_2), & j = 1, \end{cases}$$

$$\partial_1^+ u(i,j) \equiv \begin{cases} u(i+1,j) - u(i,j), & 1 \le i < N_1, \\ u(1,j) - u(N_1,j), & i = N_1, \end{cases}$$

$$\partial_2^+ u(i,j) \equiv \begin{cases} u(i,j+1) - u(i,j), & 1 \le j < N_2, \\ u(i,1) - u(i,N_2), & j = N_2. \end{cases}$$

We also define the discrete forward(+) and backward(−) gradient operator $\nabla^\pm : X \to Y$:

$$\nabla^\pm u(i,j) \equiv \left(\partial_1^\pm u(i,j), \partial_2^\pm u(i,j)\right).$$

Considering inner products on X and Y, the corresponding discrete backward(−) and forward(+) adjoint operator $\mathrm{div}^\mp : Y \to X$ of $-\nabla^\pm$ is obtained:

$$\mathrm{div}^\mp\, p(i,j) \equiv \partial_1^\mp p_1(i,j) + \partial_2^\mp p_2(i,j).$$

These operators are naturally extended on $\mathbf{X}$ and $\mathbf{Y}$.

$$\begin{array}{rcl} \nabla^\pm : \mathbf{X} \to \mathbf{Y} & \text{by} & \nabla \mathbf{u} = (\nabla^\pm u_1, \ldots, \nabla^\pm u_k), \\ \mathrm{div}^\pm : \mathbf{Y} \to \mathbf{X} & \text{by} & \mathrm{div}^\pm\, \mathbf{p} = (\mathrm{div}^\pm\, p^1, \ldots, \mathrm{div}^\pm\, p^k). \end{array}$$

Note that we have $(\mathbf{p}, -\nabla^\pm \mathbf{u})_{\mathbf{Y}} = (\mathrm{div}^\mp\, \mathbf{p}, \mathbf{u})_{\mathbf{X}}$. In the rest of paper, $k = 2$ is fixed.

3.2 Algorithm for the first step (2.4)

In the first step (2.4), we have four models: TVS-L^p, H^1S-L^p, VTV-L^p, and VH^1-L^p ($p \ge 1$). First of all, we introduce our algorithm for solving the TVS-L^p model. Since the H^1 regularization or the absence of divergence free constraint in the general model (2.4) are easily handled with reduced number of Lagrangian multipliers in the proposed algorithm, it is redundant to present the algorithms for the H^1S-L^p, VTV-L^p, and VH^1-L^p models.

In a discrete domain $\Omega = [1, N_1] \times [1, N_2]$, the TVS-$L^p$ model is represented:

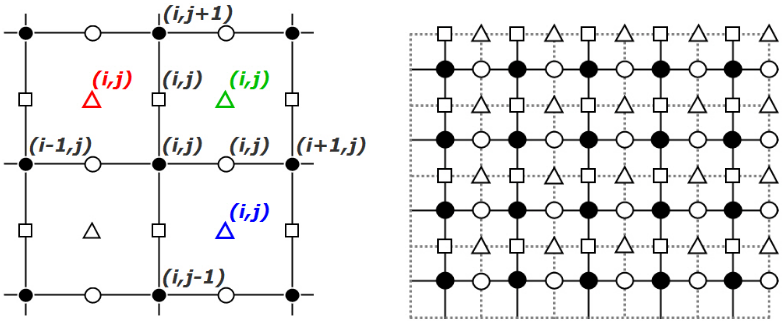
$$\min_{\substack{\mathbf{t} \in \mathbf{X} \\ \mathrm{div}^+ \mathbf{t} = 0}} \mathrm{TV}(\mathbf{t}) + \frac{\eta}{p} ||\mathbf{t} - \mathbf{t}^*||_{\Gamma,\mathbf{X}}^p, \tag{3.1}$$

where

$$\mathrm{TV}(\mathbf{t}) \equiv \sum_{(i,j)\in\Omega} \left(|\nabla^- t_1(i,j)|^2 + |\nabla^- t_2(i,j)|^2\right)^{\frac{1}{2}},$$

$$||\mathbf{t} - \mathbf{t}^*||_{\Gamma,\mathbf{X}}^p \equiv \sum_{(i,j)\in\Gamma} \left(|t_1(i,j) - t_1^*(i,j)|^p + |t_2(i,j) - t_2^*(i,j)|^p\right).$$

The choice of Γ depends on applications. We use $\Gamma = \Omega$ in image denoising, direction denoising, and image decomposition. In image inpainting, $\Gamma = \Omega \setminus \mathcal{R}$ is used where $\mathcal{R}$ is the inpainting domain. In surface reconstruction from sparse gradient, Γ indicates the location of given vector field $\mathbf{t}^*$.



(a) (b)

Figure 3.1: (a) is the rule of indexing variables, $\mathbf{p}$, $\mathbf{t}$, $\mathbf{s}$, λ_r, λ_d, and λ_f in the augmented Lagrangian functional (3.2). (b) is an example of computational domain whose size is 5×4. Note that the $\bullet$-nodes indicate the center of pixels of an image in image denoising and image inpainting.

In order to efficiently solve (3.1), we change it into a constraint minimization problem by introducing a new variable $\mathbf{p}$ and employing an operator splitting technique which is realized by a new variable $\mathbf{s}$:

$$\min \left\{ \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j)| + \frac{\eta}{p}||\mathbf{s} - \mathbf{t}^*||^p_{\Gamma,\mathbf{X}} \;\middle|\; \mathbf{p} = \nabla^- \mathbf{t},\ \mathbf{s} = \mathbf{t},\ \operatorname{div}^+ \mathbf{t} = 0 \right\}.$$

Now, we use the augmented Lagrangian method [39,75] to solve the constraint minimization problem. We firstly defined the augmented Lagrangian functional:

$$\begin{aligned}\mathcal{L}(\mathbf{t}, \mathbf{p}, \mathbf{s}; \lambda_r, \lambda_d, \lambda_f) \equiv & \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j)| + (\lambda_r, \mathbf{p} - \nabla^- \mathbf{t})_{\mathbf{Y}} + \frac{c_r}{2}||\mathbf{p} - \nabla^- \mathbf{t}||^2_{\mathbf{Y}} \\ & + \frac{\eta}{p}||\mathbf{s} - \mathbf{t}^*||^p_{\Gamma,\mathbf{X}} + (\lambda_f, \mathbf{s} - \mathbf{t})_{\mathbf{X}} + \frac{c_f}{2}||\mathbf{s} - \mathbf{t}||^2_{\mathbf{X}} \\ & + (\lambda_d, \operatorname{div}^+ \mathbf{t})_X + \frac{c_d}{2}||\operatorname{div}^+ \mathbf{t}||^2_X,\end{aligned} \tag{3.2}$$

where c_r, c_f, and c_d are positive penalty parameters and $\lambda_r \in \mathbf{Y}$, $\lambda_f \in \mathbf{X}$, and $\lambda_d \in X$ are the Lagrangian multipliers. We apply an algorithm in [39, 75] to solve the saddle-point problem of the augmented Lagrangian functional (3.2):

Step I-a. Initialize $\mathbf{t}^{(0)}$, $\mathbf{p}^{(0)}$, $\mathbf{s}^{(0)}$, $\lambda_r^{(0)}$, $\lambda_d^{(0)}$, and $\lambda_f^{(0)}$.

Step I-b. For $n \geq 0$, find $(\mathbf{t}^{(n)}, \mathbf{p}^{(n)}, \mathbf{s}^{(n)}) \simeq \underset{(\mathbf{t},\mathbf{p},\mathbf{s})\in\mathbf{X}\times\mathbf{Y}\times\mathbf{X}}{\arg\min} \mathcal{L}(\mathbf{t}, \mathbf{p}, \mathbf{s}; \lambda_r^{(n)}, \lambda_d^{(n)}, \lambda_f^{(n)})$.

Step I-c. Update $\lambda_r^{(n+1)}$, $\lambda_d^{(n+1)}$, and $\lambda_f^{(n+1)}$ as follows:

$$\begin{cases} \lambda_r^{(n+1)} = \lambda_r^{(n)} + c_r(\mathbf{p}^{(n)} - \nabla^- \mathbf{t}^{(n)}), \\ \lambda_d^{(n+1)} = \lambda_d^{(n)} + c_d(\operatorname{div}^+ \mathbf{t}^{(n)}), \\ \lambda_f^{(n+1)} = \lambda_f^{(n)} + c_f(\mathbf{s}^{(n)} - \mathbf{t}^{(n)}). \end{cases}$$

A detail discretization of each step is done on the staggered grid system in Figure 3.1. The variables, $\mathbf{p}$, $\mathbf{t}$, $\mathbf{s}$, λ_r, λ_d, and λ_f in the augmented Lagrangian functional (3.2) are defined on

$\Omega = [1, N_1] \times [1, N_2]$. In the staggered grid system, we use physically different location to evaluate the value of variables. That is, the first and second component of $\mathbf{t}$, $\mathbf{s}$, and λ_f are defined at $\square$ and $\circ$, respectively, in Figure 3.1-(a). p_1^2, p_2^1, $\lambda_{r1}^{\ 2}$, and $\lambda_{r2}^{\ 1}$ are defined at $\bullet$. Note that $\bullet$-nodes represent positions of pixels in an image. The other variables are defined at $\triangle$, but the coordinate (i, j) indicates different position. More precisely, $\lambda_d(i, j)$ is at the green triangle, $p_1^1(i, j)$ and $\lambda_{r1}^{\ 1}(i, j)$ are at the red triangle, and $p_2^2(i, j)$ and $\lambda_{r2}^{\ 2}(i, j)$ are at the blue triangle. These rules of indexing will be more reasonable when the Euler-Lagrange equations are discretized in Section 3.2.1. The periodic boundary condition is used for all variables. An example whose discrete domain is $[1, 5] \times [1, 4]$ is shown in Figure 3.1-(b).

All variables, $\mathbf{t}^{(0)}$, $\mathbf{p}^{(0)}$, $\mathbf{s}^{(0)}$, $\lambda_r^{(0)}$, $\lambda_d^{(0)}$, and $\lambda_f^{(0)}$ are initialized to be zero on the computational domain. In case of surface reconstruction from a spare gradient, since a given vector field $\mathbf{t}^*$ is defined on a piecewise smooth curve γ in $\mathbf{R}^2$, the grid points whose location indicate the curve γ and $\mathbf{t}^{(0)}|_\gamma$ should be approximately assigned. We find the nearest grid point from a given curve and copy the same vector from $\mathbf{t}^*$ on the curve.

In the Step I-b, we need to find the minimizers $\mathbf{t}^{(n)}$, $\mathbf{p}^{(n)}$, and $\mathbf{s}^{(n)}$ of the functional $\mathcal{L}$ with the fixed variables $\lambda_r^{(n)}$, $\lambda_d^{(n)}$, and $\lambda_f^{(n)}$ $(n \geq 0)$. They are approximately obtained by the alternating minimization method which consists of three sub-steps as follows:

Step I-b-1. For fixed $\mathbf{p}$ and $\mathbf{s}$, solve the minimization problem, $\min\limits_{\mathbf{t}\in\mathbf{X}} \mathcal{E}_{\mathbf{p},\mathbf{s}}(\mathbf{t})$, where

$$\begin{aligned}\mathcal{E}_{\mathbf{p},\mathbf{s}}(\mathbf{t}) \equiv (\lambda_r^{(n)}, -\nabla^-\mathbf{t})_{\mathbf{Y}} + (\lambda_d^{(n)}, \operatorname{div}^+\mathbf{t})_X + (\lambda_f^{(n)}, -\mathbf{t})_{\mathbf{X}} \\ + \frac{c_r}{2}||\mathbf{p} - \nabla^-\mathbf{t}||_{\mathbf{Y}}^2 + \frac{c_d}{2}||\operatorname{div}^+\mathbf{t}||_X^2 + \frac{c_f}{2}||\mathbf{s} - \mathbf{t}||_{\mathbf{X}}^2.\end{aligned} \tag{3.3}$$

Step I-b-2. For fixed $\mathbf{p}$ and $\mathbf{t}$, solve the minimization problem, $\min\limits_{\mathbf{s}\in\mathbf{X}} \mathcal{E}_{\mathbf{p},\mathbf{t}}(\mathbf{s})$, where

$$\mathcal{E}_{\mathbf{p},\mathbf{t}}(\mathbf{s}) \equiv \frac{\eta}{p}||\mathbf{s} - \mathbf{t}^*||_{\Gamma,\mathbf{X}}^p + (\lambda_f^{(n)}, \mathbf{s})_{\mathbf{X}} + \frac{c_f}{2}||\mathbf{s} - \mathbf{t}||_{\mathbf{X}}^2. \tag{3.4}$$

Step I-b-3. For fixed $\mathbf{s}$ and $\mathbf{t}$, solve the minimization problem, $\min\limits_{\mathbf{p}\in\mathbf{Y}} \mathcal{E}_{\mathbf{s},\mathbf{t}}(\mathbf{p})$, where

$$\mathcal{E}_{\mathbf{s},\mathbf{t}}(\mathbf{p}) \equiv \sum_{(i,j)\in\Omega} |\mathbf{p}(i, j)| + (\lambda_r^{(n)}, \mathbf{p})_{\mathbf{Y}} + \frac{c_r}{2}||\mathbf{p} - \nabla^-\mathbf{t}||_{\mathbf{Y}}^2. \tag{3.5}$$

The periodic boundary condition is enforced in the variables $\mathbf{t}$, $\mathbf{p}$, and $\mathbf{s}$ after each step and the updated variable is used in the next step. We numerically observe that a single iteration of the alternating minimization scheme is enough to solve the saddle-point problem of the augmented Lagrangian functional (3.2). In [39,40], it is mathematically proved that a single iteration is enough to generate a convergent sequence in the ROF model, the vectorial TV model, and high order models. In every iteration in Step I, the relative L^1 errors of each component of all Lagrangian multipliers are measured. If all errors are less than a given error bound ϵ_1, we stop the iteration.

In the rest of subsection, we explain the detail algorithm to find the minimizers in (3.3), (3.4), and (3.5) on the staggered grid system.

3.2.1 Step I-b-1: Minimization of the energy $\mathcal{E}_{\mathbf{p},\mathbf{s}}(\cdot)$

For the fixed variables $\mathbf{p}$ and $\mathbf{s}$, we obtain the minimizer of the energy functional (3.3). The Euler-Lagrange equation of $\mathcal{E}_{\mathbf{p},\mathbf{s}}(\cdot)$ yields coupled PDEs:

$$\operatorname{div}^+\lambda_r^{(n)} - \nabla^-\lambda_d^{(n)} - \lambda_f^{(n)} + c_r \operatorname{div}^+(\mathbf{p} - \nabla^-\mathbf{t}) - c_d\nabla^-(\operatorname{div}^+\mathbf{t}) - c_f(\mathbf{s} - \mathbf{t}) = \mathbf{0}.$$

Using the notation in Section 3.1, the above PDEs are written as

$$
\begin{aligned}
-(c_r + c_d)\partial_1^+\partial_1^- t_1 - c_r\partial_2^+\partial_2^- t_1 - c_d\partial_1^-\partial_2^+ t_2 + c_f t_1 = \\
- \partial_1^+\lambda_r^{(n)}{}_1^1 - \partial_2^+\lambda_r^{(n)}{}_2^1 + \partial_1^-\lambda_d^{(n)} + \lambda_{f1}^{(n)} - c_r\partial_1^+ p_1^1 - c_r\partial_2^+ p_2^1 + c_f s_1,
\end{aligned} \tag{3.6}
$$

$$
\begin{aligned}
-(c_r + c_d)\partial_2^+\partial_2^- t_2 - c_r\partial_1^+\partial_1^- t_2 - c_d\partial_2^-\partial_1^+ t_1 + c_f t_2 = \\
- \partial_1^+\lambda_r^{(n)}{}_1^2 - \partial_2^+\lambda_r^{(n)}{}_2^2 + \partial_2^-\lambda_d^{(n)} + \lambda_{f2}^{(n)} - c_r\partial_1^+ p_1^2 - c_r\partial_2^+ p_2^2 + c_f s_2.
\end{aligned} \tag{3.7}
$$

We use the rule of indexing variables in Figure 3.1. Introducing the identity operator $\mathcal{I}f(i,j) = f(i,j)$ and shifting operators,

$$\mathcal{S}_1^\pm f(i,j) = f(i\pm 1, j) \quad \text{and} \quad \mathcal{S}_2^\pm f(i,j) = f(i, j\pm 1),$$

the discretization of (3.6) at $\square$-nodes and (3.7) at $\circ$-nodes is written as

$$
\begin{aligned}
\left(-(c_r + c_d)\left(\mathcal{S}_1^+ - 2\mathcal{I} + \mathcal{S}_1^-\right) - c_r\left(\mathcal{S}_2^+ - 2\mathcal{I} + \mathcal{S}_2^-\right) + c_f\mathcal{I}\right) t_1(i,j) \\
- c_d\left(\mathcal{S}_2^+ - \mathcal{I} - \mathcal{S}_1^-\mathcal{S}_2^+ + \mathcal{S}_1^-\right) t_2(i,j) = f_1(i,j),
\end{aligned} \tag{3.8}
$$

where

$$
\begin{aligned}
f_1(i,j) = &- \left[\left(\mathcal{S}_1^+ - \mathcal{I}\right)\left(\lambda_r^{(n)}{}_1^1 + c_r p_1^1\right) + \left(\mathcal{S}_2^+ - \mathcal{I}\right)\left(\lambda_r^{(n)}{}_2^1 + c_r p_2^1\right)\right](i,j) \\
&+ \left[\left(\mathcal{I} - \mathcal{S}_1^-\right)\lambda_d^{(n)} + \lambda_{f1}^{(n)} - c_r\left(\mathcal{S}_2^+ - \mathcal{I}\right) p_2^1 + c_f s_1\right](i,j)
\end{aligned}
$$

and

$$
\begin{aligned}
\left(-(c_r + c_d)\left(\mathcal{S}_2^+ - 2\mathcal{I} + \mathcal{S}_2^-\right) - c_r\left(\mathcal{S}_1^+ - 2\mathcal{I} + \mathcal{S}_1^-\right) + c_f\mathcal{I}\right) t_2(i,j) \\
- c_d\left(\mathcal{S}_1^+ - \mathcal{S}_1^+\mathcal{S}_2^- - \mathcal{I} + \mathcal{S}_2^-\right) t_1(i,j) = f_2(i,j),
\end{aligned} \tag{3.9}
$$

where

$$
\begin{aligned}
f_2(i,j) = &- \left[\left(\mathcal{S}_2^+ - \mathcal{I}\right)\left(\lambda_r^{(n)}{}_2^2 + c_r p_2^2\right) + \left(\mathcal{S}_1^+ - \mathcal{I}\right)\left(\lambda_r^{(n)}{}_1^2 + c_r p_1^2\right)\right](i,j) \\
&+ \left[\left(\mathcal{I} - \mathcal{S}_2^-\right)\lambda_d^{(n)} + \lambda_{f2}^{(n)} - c_r\left(\mathcal{S}_1^+ - \mathcal{I}\right) p_2^1 + c_f s_1\right](i,j).
\end{aligned}
$$

Since the variable $\mathbf{t}$ is periodically extended, we apply the discrete Fourier transform $\mathcal{F}$ to solve (3.8) and (3.9). The shifting operators are essentially discrete convolutions and then their discrete Fourier transforms are the componentwise multiplication in the frequency domain. For representing discrete frequency, u_i and u_j, we have

$$\mathcal{F}\mathcal{S}_1^\pm f(u_i,u_j) = e^{\pm\sqrt{-1}v_i}\mathcal{F}f(u_i,u_j) \qquad \mathcal{F}\mathcal{S}_2^\pm f(u_i,u_j) = e^{\pm\sqrt{-1}v_j}\mathcal{F}f(u_i,u_j)$$

where

$$v_i = \frac{2\pi}{N_1}u_i,\ u_i = 1, \cdots, N_1, \quad \text{and} \quad v_j = \frac{2\pi}{N_2}u_j,\ u_j = 1, \cdots, N_2,$$

It yields a system of linear equations:

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}\begin{pmatrix} \mathcal{F}t_1(u_i,u_j) \\ \mathcal{F}t_2(u_i,u_j) \end{pmatrix} = \begin{pmatrix} \mathcal{F}f_1(u_i,u_j) \\ \mathcal{F}f_2(u_i,u_j) \end{pmatrix},$$

where the coefficients are

$$
\begin{aligned}
a_{11} &= -2(c_r + c_d)(\cos v_i - 1) - 2c_r(\cos v_j - 1) + c_f, \\
a_{12} &= -c_d\left(1 - \cos v_j + \sqrt{-1}\sin v_j\right)\left(\cos v_i - 1 + \sqrt{-1}\sin v_i\right), \\
a_{21} &= -c_d\left(1 - \cos v_i + \sqrt{-1}\sin v_i\right)\left(\cos v_j - 1 + \sqrt{-1}\sin v_j\right), \\
a_{22} &= -2c_r(\cos v_i - 1) - 2(c_r + c_d)(\cos v_j - 1) + c_f.
\end{aligned}
$$

Now, we have N_1N_2 numbers of 2×2 systems. The determinant of the coefficients matrix for all discrete frequencies is not zero because the penalty parameters c_r, c_d, and c_f are positive:

$$D \equiv \det\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = (2c_r(\cos v_i + \cos v_j - 2) - c_f)(2(c_r+c_d)(\cos v_i + \cos v_j - 2) - c_f).$$

After the systems of linear equations are solved for each frequency, we use the discrete inverse Fourier transform to obtain t_1 and t_2:

$$t_1 = \Re\left(\mathcal{F}^{-1}\left(\frac{a_{22}\mathcal{F}f_1 - a_{12}\mathcal{F}f_2}{D}\right)\right), \quad t_2 = \Re\left(\mathcal{F}^{-1}\left(\frac{-a_{21}\mathcal{F}f_1 + a_{11}\mathcal{F}f_2}{D}\right)\right),$$

where $\Re(\cdot)$ is the real part of a complex number.

3.2.2 Step I-b-2: Minimization of the energy $\mathcal{E}_{\mathbf{p},\mathbf{t}}(\cdot)$

In order to obtain the minimizer of energy functional (3.4), we divide $\mathcal{E}_{\mathbf{p},\mathbf{t}}(\cdot)$ into two parts:

$$\mathcal{E}_{\mathbf{p},\mathbf{t}}(\mathbf{s}) = \mathcal{E}_{\Omega\backslash\Gamma}(\mathbf{s}) + \mathcal{E}_{\Gamma}(\mathbf{s}),$$

where

$$\mathcal{E}_{\Omega\backslash\Gamma}(\mathbf{s}) \equiv \sum_{(i,j)\in\Omega\backslash\Gamma}\sum_{l=1}^{2}\left(\lambda_{fl}^{(n)}(i,j)s_l(i,j) + \frac{c_f}{2}|s_l(i,j) - t_l(i,j)|^2\right),$$

$$\mathcal{E}_{\Gamma}(\mathbf{s}) \equiv \sum_{(i,j)\in\Gamma}\sum_{l=1}^{2}\left(\frac{\eta}{p}|s_l(i,j) - t_l^*(i,j)|^p + \lambda_{fl}^{(n)}(i,j)s_l(i,j) + \frac{c_f}{2}|s_l(i,j) - t_l(i,j)|^2\right).$$

The minimizer in the first energy functional $E_{\Omega\backslash\Gamma}(\cdot)$ is easily obtained because it is a quadratic polynomial for each coordinate (i,j). For the second energy functional, we reformulate it as follows:

$$\mathcal{E}_{\Gamma}(\mathbf{s}) \equiv \sum_{(i,j)\in\Gamma}\sum_{l=1}^{2}\left(\frac{\eta}{p}|s_l(i,j) - t_l^*(i,j)|^p + \frac{c_f}{2}\left|s_l(i,j) - t_l(i,j) + \frac{1}{c_f}\lambda_{fl}^{(n)}(i,j)\right|^2\right) + C,$$

where C does not count on the minimization. For each coordinate $(i,j)\in\Gamma$ and the index $l=1$ and 2, letting

$$x(i,j) = s_l(i,j) - t_l^*(i,j) \quad \text{and} \quad x_0(i,j) = -t_l^*(i,j) + t_l(i,j) - \frac{1}{c_f}\lambda_{fl}^{(n)}(i,j),$$

the problem of finding the minimizer in $E_\Gamma(\cdot)$ is changed to find the minimizer of the function

$$f(x) = \frac{\eta}{p}|x|^p + \frac{c_f}{2}|x - x_0|^2.$$

By using the fixed point theorem to $f'(x)$, it is straightforward to prove that the minimizer assumes $x = \alpha x_0$, $0 \le \alpha \le 1$. Now, we find α such that $f(\alpha x_0)$ is minimized. It yields a problem of finding a minimizer of a radical function $g(\alpha) = \frac{\eta}{p}|x_0|^p\alpha^p + \frac{c_f}{2}|x_0|^2(\alpha-1)^2$. When $p>1$ and $p\ne 2$, we use the fixed point theorem to find a root of $g'(\alpha)=0$. When $p=2$, we have the analytical solution of $\alpha(i,j) = \frac{c_f}{\eta + c_f}$. When $p=1$, we have $\alpha(i,j) = \max\left(0, 1 - \frac{\eta}{c_f|x_0(i,j)|}\right)$ because $\alpha \ge 0$; see also [41]. Therefore, the minimizer of $E_{\mathbf{p},\mathbf{t}}(\mathbf{s})$ is represented by

$$(i,j)\notin\Gamma \quad \Rightarrow \quad s_l(i,j) = t_l(i,j) - \frac{1}{c_f}\lambda_{fl}^{(n)}(i,j),$$

$$(i,j)\in\Gamma \quad \Rightarrow \quad s_l(i,j) = t_l^*(i,j) + \alpha(i,j)x_0(i,j),$$

where $l=1$ and 2.

3.2.3 Step I-b-3: Minimization of the energy $\mathcal{E}_{\mathbf{s},\mathbf{t}}(\cdot)$

We apply the same approach in [41] to find the closed form of the minimizer of the functional (3.5). By completing the square, the energy functional $\mathcal{E}_{\mathbf{s},\mathbf{t}}(\cdot)$ in (3.5) can be written as:

$$\mathcal{E}_{\mathbf{s},\mathbf{t}}(\mathbf{p}) = \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j)| + \frac{c_r}{2} \left|\left| \mathbf{p} - \left(\nabla^- \mathbf{t} - \frac{1}{c_r} \lambda_r{}^{(n)} \right) \right|\right|_{\mathbf{Y}}^2 + C,$$

where C does not count on the minimization. The close form solution is obtained by

$$\mathbf{p}(i,j) = \max \left\{ 0, 1 - \frac{1}{c_r |\mathbf{w}(i,j)|} \right\} \mathbf{w}(i,j),$$

where

$$\mathbf{w} = \nabla^- \mathbf{t} - \frac{1}{c_r} \lambda_r{}^{(n)}.$$

More details are shown in [41].

3.3 Algorithm for the second step (2.5)

After the regularized tangent vector field $\mathbf{t}$ is obtained from the first step, the regularized normal vector field $\mathbf{n}$ is defined by $\mathbf{t}^\perp$. In the second step (2.5), we have two models: TV$\mathbf{n}$-L^p and $H^1\mathbf{n}$-L^p ($p \geq 1$). We introduce the algorithm for solving the TV$\mathbf{n}$-L^p model. The algorithm for the $H^1\mathbf{n}$-L^p is easily obtained with reduced number of Lagrangian multipliers from the algorithm in this subsection. Note that we do not need to use the augmented Lagrangian method to solve the $H^1\mathbf{n}$-L^2 model with $\Gamma = \Omega$. However, we still need to use the proposed algorithm to solve the $H^1\mathbf{n}$-L^2 model with $\Gamma \subsetneq \Omega$.

In the discrete domain $\Omega = [1, N_1] \times [1, N_2]$, the TV$\mathbf{n}$-$L^p$ model ($p \geq 1$) is represented to find a minimizer in $\mathbf{X}$:

$$\min_{I \in \mathbf{X}} \mathrm{TV}_{\mathbf{n}}(I) + \frac{\xi}{p} ||I - I^*||_{\Gamma,\mathbf{X}}^p, \tag{3.10}$$

where

$$\mathrm{TV}_{\mathbf{n}}(I) \equiv \sum_{(i,j)\in\Omega} |\nabla^+ I(i,j) - \mathbf{n}(i,j)|,$$
$$||I - I^*||_{\Gamma,\mathbf{X}}^p \equiv \sum_{(i,j)\in\Gamma} |I(i,j) - I^*(i,j)|^p.$$

In order to efficiently solve (3.10), we change it into a constraint minimization problem by introducing a new variable $\mathbf{p}$ and employing an operator splitting technique which is realized by a new variable J:

$$\min \left\{ \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j) - \mathbf{n}(i,j)| + \frac{\xi}{p} ||J - I^*||_{\Gamma,\mathbf{X}}^p \;\middle|\; \mathbf{p} = \nabla^+ I, \ J = I \right\}.$$

Now, we use the augmented Lagrangian method [39,75] to solve the constraint minimization problem via the augmented Lagrangian functional:

$$\begin{aligned} \mathcal{L}(\mathbf{p}, I, J; \lambda_r, \lambda_f) \equiv & \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j) - \mathbf{n}(i,j)| + \frac{\xi}{p} ||J - I^*||_{\Gamma,X}^p \\ & + (\lambda_r, \mathbf{p} - \nabla^+ I)_Y + (\lambda_f, J - I)_X + \frac{c_r}{2} ||\mathbf{p} - \nabla^+ I||_Y^2 + \frac{c_f}{2} ||J - I||_X^2, \end{aligned} \tag{3.11}$$

where c_r and c_f are positive penalty parameters and $\lambda_r \in \mathbf{Y}$ and $\lambda_f \in \mathbf{X}$ are the Lagrangian multipliers. Note that, even though $p = 2$, the auxiliary variable J should be used in image inpainting and surface reconstruction from sparse gradient because Γ is a proper subset of Ω. We apply an algorithm in [39,75] to solve the saddle-point problem of the augmented Lagrangian functional (3.11):

Step II-a. Initialize $\mathbf{p}^{(0)}$, $I^{(0)}$, $J^{(0)}$, ${\lambda_r}^{(0)}$, and ${\lambda_f}^{(0)}$.

Step II-b. For $n \geq 0$, find $(\mathbf{p}^{(n)}, I^{(n)}, J^{(n)}) \simeq \underset{(\mathbf{p},I,J)}{\arg\min}\, \mathcal{L}(\mathbf{p}, I, J; {\lambda_r}^{(n)}, {\lambda_f}^{(n)})$.

Step II-c. Update ${\lambda_r}^{(n+1)}$ and ${\lambda_f}^{(n+1)}$ as follows:

$$\begin{cases} {\lambda_r}^{(n+1)} = {\lambda_r}^{(n)} + c_r(\mathbf{p}^n - \nabla^+ I^{(n)}), \\ {\lambda_f}^{(n+1)} = {\lambda_f}^{(n)} + c_d(J^{(n)} - I^{(n)}). \end{cases}$$

In the Step II-b, we approximately obtain the minimizer $(\mathbf{p}^{(n)}, I^{(n)}, J^{(n)})$ in the functional $\mathcal{L}$ by using the alternating minimization method as follows:

Step II-b-1. For fixed $\mathbf{p}$ and J, solve the minimization problem, $\min\limits_{I \in \mathbf{X}} \mathcal{E}_{\mathbf{p},J}(I)$, where

$$\mathcal{E}_{\mathbf{p},J}(I) \equiv ({\lambda_r}^{(n)}, -\nabla^+ I)_Y + ({\lambda_f}^{(n)}, -I)_X + \frac{c_r}{2}||\mathbf{p} - \nabla^+ I||_Y^2 + \frac{c_f}{2}||J - I||_X^2. \quad (3.12)$$

Step II-b-2. For fixed $\mathbf{p}$ and I, solve the minimization problem, $\min\limits_{J \in \mathbf{X}} \mathcal{E}_{\mathbf{p},I}(J)$, where

$$\mathcal{E}_{\mathbf{p},I}(J) \equiv \frac{\eta}{p}||J - I^*||_{\Gamma,X}^p + ({\lambda_f}^{(n)}, J)_X + \frac{c_f}{2}||J - I||_X^2. \quad (3.13)$$

Step II-b-3. For fixed I and J, solve the minimization problem, $\min\limits_{\mathbf{p} \in Y} \mathcal{E}_{I,J}(\mathbf{p})$, where

$$\mathcal{E}_{I,J}(\mathbf{p}) \equiv \sum_{(i,j)\in\Omega} |\mathbf{p}(i,j) - \mathbf{n}(i,j)| + (\lambda_r, \mathbf{p})_Y + \frac{c_r}{2}||\mathbf{p} - \nabla^+ I||_Y^2. \quad (3.14)$$

The periodic boundary condition is enforced in the variables $\mathbf{p}$, I, and J after each step and the updated variable is used in the next step. We numerically observe that a single iteration of the alternating minimization scheme is enough to solve the saddle-point problem of the augmented Lagrangian functional (3.11). In every iteration in Step II, the relative L^1 errors of each component of all Lagrangian multipliers are measured. If all errors are less than a given error bound ϵ_2, we stop the iteration.

The detail algorithms to find the minimizer in the functionals (3.12), (3.13), and (3.14) are straightforward. In Step II-b-1, the Euler-Lagrange equation of the energy functional (3.12) is

$$\operatorname{div}^- {\lambda_r}^{(n)} - {\lambda_f}^{(n)} + c_r \operatorname{div}^-(\mathbf{p} - \nabla^+ I) - c_f(J - I) = 0.$$

It can be solved by the discrete Fourier transform in a similar way in Step I-b-1. The minimizers in Step II-b-2 and Step II-b-3 are obtained by using the same methods in Step I-b-2 and Step I-b-3.

3.4 Computational time

The computational time of the proposed augmented Lagrangian method (ALM) is compared with the dual formulation (DUAL) of TV-Stokes equation for image denoising in [26, 46] and the additive operator splitting (AOS) scheme for image inpainting in [74, 76]

(a) (b)

image	size	first step		second step	
		ALM	AOS	ALM	AOS
(a)	3711	1.70	44.80	0.96	26.69
(b)	14258	37.64	756.72	9.23	409.58

Table 3.1: The computational time in seconds of the proposed ALM is compared with the AOS scheme for image inpainting in [74]. The size indicates the number of pixels in inpainting domain which is the red region in the left side of (a) and (b). The right side of (a) and (b) is the result of the TVS-L^1+TV$\mathbf{n}$-L^2 model for image inpainting.

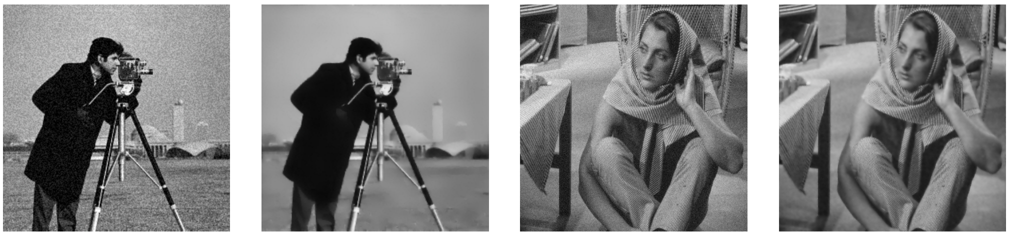
In Table 3.1, the computational time of the proposed ALM is compared with the AOS scheme for same examples in image inpainting in [74]. We use the TVS-L^1+TV$\mathbf{n}$-L^2 model and the results are shown above the table. Even though the model has a nonlinearity of the fidelity term in the first step, the computational time of the proposed ALM is much smaller than the AOS scheme. The test system is Intel(R) Core(TM)2 Duo CPU P8600 2.4GHz 32-bit processor and 3GB RAM. The same parameters in [74] are used for the AOS scheme. In (a) and (b), we use $c_r = 10$, $c_d = 1$, $c_f = 1$, and $\eta = 1$ and $\epsilon_1 = 10^{-2}$ for the TVS-L^1 model and $c_r = 10$, $c_f = 1$, $\xi = 1$ and $\epsilon_2 = 10^{-2}$ for the TV$\mathbf{n}$-L^2 model.

In Table 3.2, the proposed method is compared with the dual formulation of TV-Stokes equations for image denoising [46]. We take their computational time in the table and the test system used in [46] is a 2 Opteron 270 dualcore 64-bit processor and 8GB RAM. We use TVS-L^2+TV$\mathbf{n}$-L^2 model and the results are shown above the table. In (a), we use $c_r = 10$, $c_d = 1$, $c_f = 1$, $\eta = 10$ and $\epsilon_1 = 10^{-2}$ for the TVS-L^2 model and $c_r = 10$, $\xi = 10$ and $\epsilon_2 = 10^{-2}$ for the TV$\mathbf{n}$-L^2 model. In (b), we use $c_r = 1$, $c_d = 10^{-2}$, $c_f = 1$, $\eta = 15$, and $\epsilon_1 = 10^{-2}$ for the TVS-L^2 model and $c_r = 1$, $\xi = 15$, and $\epsilon_2 = 10^{-2}$ for the TV$\mathbf{n}$-L^2 model. Through a discussion with the authors in [46], we figure out that a single thread in the Opteron system is used. We expect that the computational cost between the dual formulation and the proposed algorithm for the TV-Stokes equation may not be quite different under the same system.

4 Applications

In this section, we show the proposed model can be used for a number of applications: image inpainting, image decomposition, surface reconstruction from sparse gradient, direction denoising, and image denoising. Moreover, we numerically investigate various effects of different norms in the proposed general models: the first step (2.4) and the second step (2.5).

The effects of different norms which we want to show via many examples are very straightforward. That is, the TV regularization preserves discontinuities of data and the H^1 regularization enforces continuities of data. Since the data in the first step is different from the data in second step, the comparison of norms in each step should be carefully done. We also illustrate the role of divergence free constraint in image inpainting. The difference of L^1 and L^2 fidelities in (2.4) is shown in image



(a) (b)

image	size	first step		second step	
		ALM	DUAL	ALM	DUAL
(a)	256^2	4.15	17.4	0.98	2.2
(b)	512^2	36.49	128.2	5.48	20.7

Table 3.2: The computational time in seconds of the proposed ALM is compared with the dual formulation for the TV-Stokes denoising algorithm in [46]. The right side of (a) and (b) is the result of the TVS-L^2+TV**n**-L^2 model for image denoising. The left side of (a) and (b) is the noisy image which is directly taken from [46].

decomposition and direction denoising.

We use the standard hue-saturation-value (HSV) map to represent a vector field. Since every vector field in this paper is defined on two dimensional space, the value in the HSV map is fixed to 1. For example, the white color in the HSV map presents the zero vector.

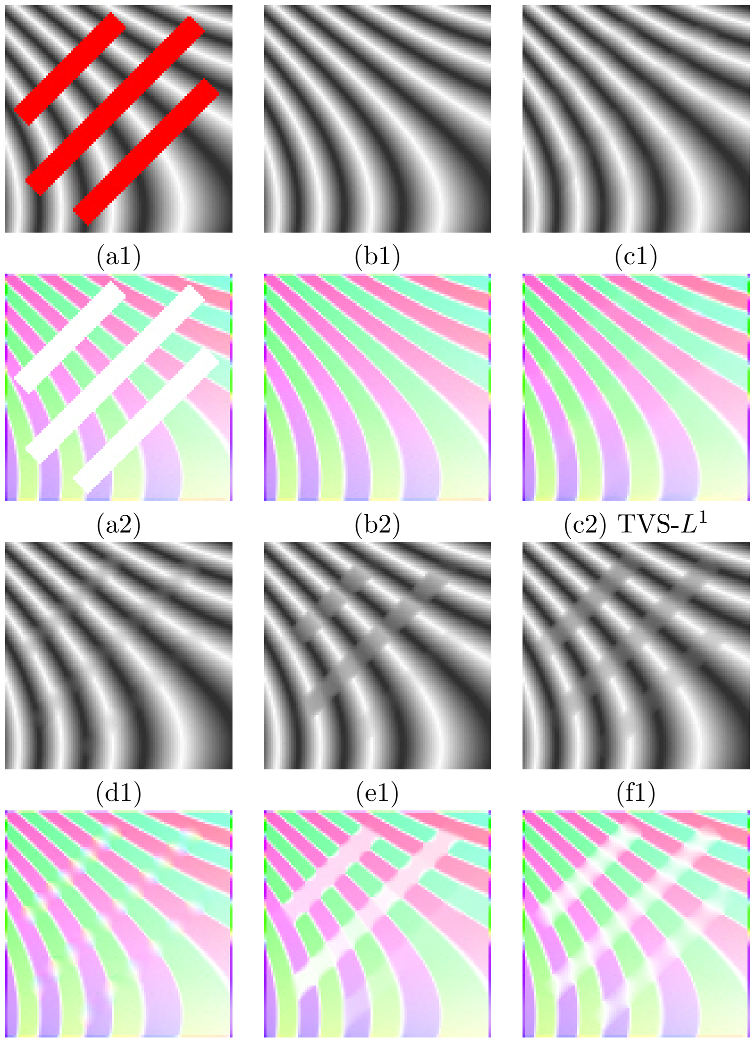
4.1 Image inpainting

In image inpainting, we recover degraded or missing parts $\mathcal{R}$ in an image domain Ω. We denote inpainting regions $\mathcal{R}$ as the red color; see Figure 4.1-(a1). A given image I^* is defined on $\Gamma = \Omega \setminus \mathcal{R}$. In order to interpolate image values I^* on the boundary of $\mathcal{R}$, we use the proposed two-step method. More precisely, a tangential vector field is interpolated by (2.4) in the first step. An inpainted image is recovered by (2.5) in the second step with the regularized normal vector field from the first step. In this subsection, we observe that the divergence free constraint is crucial and various effects of different regularizations are shown.

First of all, we show the advantage of using the divergence free constraint in image inpainting. The comparison between TV and H^1 regularization in the first step (2.4) is also illustrated. For this purpose, we numerically compare four different methods to regularize a given tangential vector field in image inpainting via different regularization and constraint:

$$\text{TVS-}L^1, \quad H^1\text{S-}L^1, \quad \text{VTV-}L^1, \quad \text{and} \quad \text{V}H^1\text{-}L^1. \tag{4.1}$$

In Figure 4.1, we fix the second step as the TV**n**-L^2 model for the fair comparison. The red regions in (a1) are inpainting domains and (b1) is the original image. (a2) and (b2) are the tangential vector field of (a1) and (b1), respectively, rendered by the HSV color map. From (c2), (d2), (e2), and (f2), we show results obtained by different regularization in (4.1). (c1), (d1), (e1), and (f1) are obtained by the TV**n**-L^2 model with different regularized vector fields $\mathbf{n}$ in (c2), (d2), (e2), and (f2), respectively. The error bound is $\epsilon_1 = 10^{-2}$ in the first step and $\epsilon_2 = 10^{-3}$ in the second step. In (c2), we use $c_r = 10$, $c_d = 10$, $c_f = 1$, and $\eta = 10$ for the TVS-L^1 model. In (d2), we use $c_f = 1$ and $\eta = 10$. In (e2), we use $c_r = 10$, $c_f = 1$, and $\eta = 10$. In (f2), we use $c_f = 1$ and $\eta = 10$. In order to reconstruct the inpainted images from the regularized vector fields, we use the TV**n**-L^2 model with $c_r = 10$, $c_f = 1$, and $\xi = 10$, in (c1), (d1), (e1), and (f1).



(a1) (b1) (c1)

(a2) (b2) (c2) TVS-L^1

(d1) (e1) (f1)

(d2) H^1S-L^1 (e2) VTV-L^1 (f2) VH^1-L^1

Figure 4.1: The comparison of different regularization in (4.1): the red regions in (a1) are inpainting domains and (b1) is the original image. (a2) and (b2) are the tangential vector field of (a1) and (b1), respectively, rendered by the HSV color map. Inpainting results are shown in (c1), (d1), (e1), and (f1). They are obtained by the different inpainted tangential vector fields in (c2), (d2), (e2), and (f2), respectively. From (c2) and (e2), we show the advantage of using the divergence free constraint. From (c2) and (d2), we show that the TV norm in (4.1) preserves the discontinuities of the image gradient but the H^1 norm smears them out.

In (c2) and (d2), the divergence free constraint in TVS-L^1 model and H^1S-L^1 model shows the advantage for propagating the vectors on the boundary of $\mathcal{R}$ into the large inpainting domains. Since the inpainting regions on the left top in (a1) are intentionally drawn to be thicker than the width of strip in (a2), the results of VTV-L^1 model shown in (e2) and VH^1-L^1 model shown in (f2) does not recover a correct tangential vector field on these regions because the divergence free constraint is not imposed. Even though the divergence free constraint is good for a large inpainting domain, we notice that the proposed model can not outperform the Euler's elastica model [47, 52] in the sense of long connectivity.

The difference between TV and H^1 regularization in (4.1) is easily observed in Figure 4.1. In case of the H^1 norm, jump discontinuities in the image gradient are not allowed in the minimizer of (2.4) since the Euler-Lagrangian equation of each component of the vector is a Poisson equation. However, the TV norm allows such jump discontinuities in the regularized tangential vector field. These phenomena are shown in (c2) and (d2). Since the discontinuities of image gradient reflect sharp ridges or valleys, we can see a good reconstruction of ridges and valleys in (c1) but they are smeared out in (d1).

Now, we show the different effect of TV and H^1 regularization in the second step (2.5). For this purpose, we numerically compare two methods to reconstruct an image via different regularization:

$$\mathrm{TV}\mathbf{n}\text{-}L^2 \quad \text{and} \quad H^1\mathbf{n}\text{-}L^2. \tag{4.2}$$

In Figures 4.2-(c) and 4.2-(e), we fix the first step as the TVS-L^1 model to obtain a regularized vector field $\mathbf{n}$ for the fair comparison. The inpainting domain in (a) includes jump discontinuities of the image and discontinuities of the image gradient. As we have seen the preservation of discontinuities of image gradient under the TVS-L^1 model in Figure 4.1, the information of sharp ridges or valleys are well preserved in the regularized vector field $\mathbf{n}$. Therefore, ridges and valleys are well recovered by two models in (4.2). The difference between the TV and H^1 norm in the second step (4.2) is shown in (c) and (e) or (f). Since the H^1 norm enforces the surface continuity in the surface reconstruction step, the edges in (e) can not be restored. The image in (f) is an extreme case. Even though we use the normal vector field $\mathbf{n} = \nabla I$ of the original image I in the $H^1\mathbf{n}$-L^2 model, the edges in (f) can not be reconstructed. In (c), the TV$\mathbf{n}$-L^2 model reconstructs sharp edges because of the TV regularization. The TV-L^2 inpainting model in (d) is good for reconstructing edges but it can not recover ridges or valleys because the TV-L^2 model is the case of $\mathbf{n} = 0$ in the TV$\mathbf{n}$-L^2 model. We use $c_r = 10$, $c_d = 10^{-3}$, $c_f = 1$, $\eta = 10$, and $\epsilon_1 = 5 \cdot 10^{-3}$ for the TVS-L^1 model in (c) and (e). The surface reconstruction in (c) and (e) uses the TV$\mathbf{n}$-L^2 model with $c_r = 10$, $c_f = 1$, $\xi = 10$, and $\epsilon_2 = 10^{-3}$ and the H^1S-L^2 model with $c_f = 1$, $\xi = 10^3$, and $\epsilon_2 = 10^{-8}$. In (f), we use $c_f = 1$, $\xi = 10$, and $\epsilon_2 = 10^{-5}$ for the H^1S-L^2 model. In (d), we use $c_r = 10$, $\xi = 10^3$, and $\epsilon_2 = 10^{-5}$.

4.2 Image decomposition

In this subsection, we show that one of generalized TV-stokes models can be applied to image decomposition introduced in [14]. The conventional meaning of image decomposition is to separate an image into a cartoon part and a texture part or a noise part [77–79]. Here, we refer to a different meaning which is a separation of jump discontinuities of an image and discontinuities of the image gradient or smooth regions. Note that we do not separately obtain texture parts from an image.

In the generalized TV-Stokes model, we use the L^1 fidelity in the first step (2.4) to show a decomposition property. More specifically, the TVS-L^1+TV$\mathbf{n}$-L^2 model decomposes an image I into two functions $I = f + g$ in (2.16): the function f has jump discontinuities of the image and the function g has discontinuities of the image gradient or smooth regions of the image. From a given image I^*, we obtain the regularized normal vector field $\mathbf{n}$ from the TVS-L^1 model. The functions g (2.14) and f (2.15) are obtained by the TV$\mathbf{n}$ model and the ROF model, respectively. The main focus in this subsection is that the TVS-L^2 model does not make a decomposition property introduced in [14] because of the L^2 fidelity

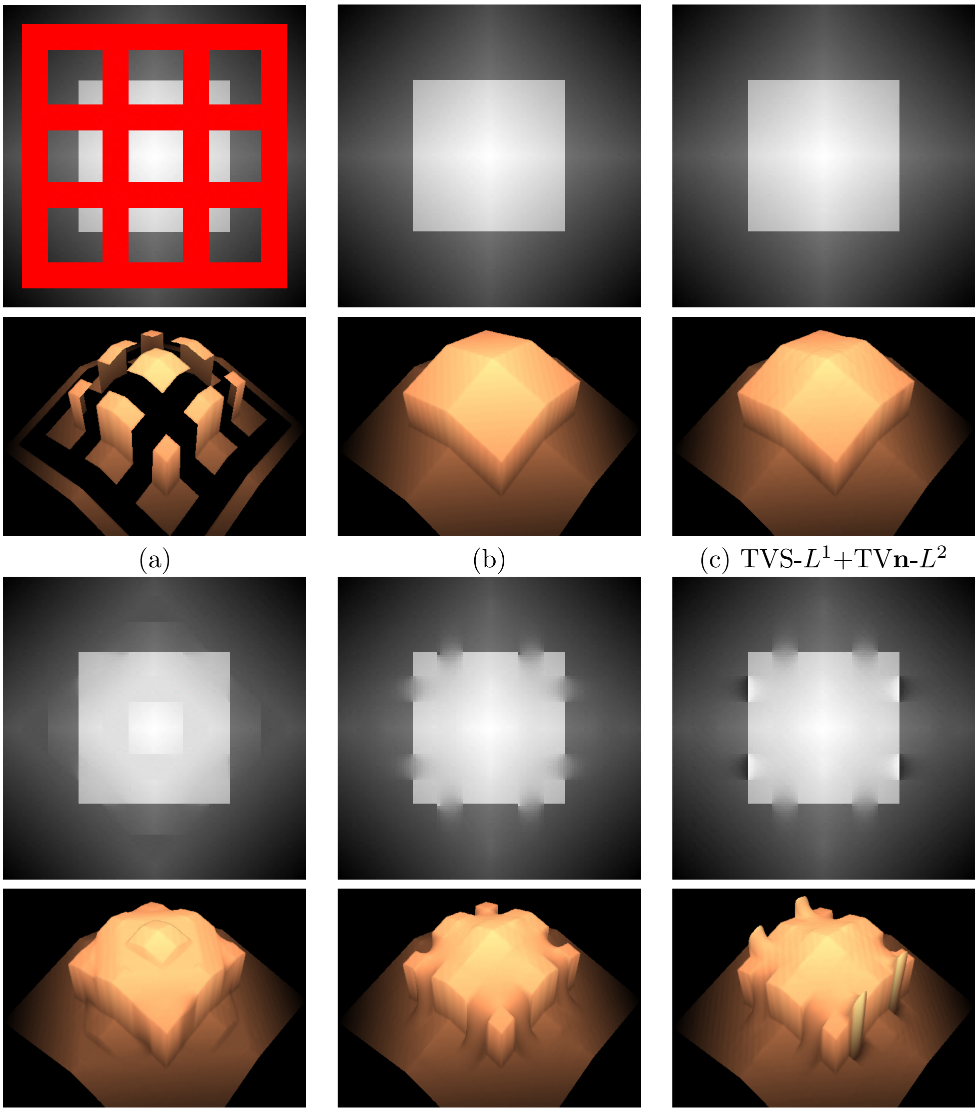

(a)
(b)
(c) TVS-L^1+TV$\mathbf{n}$-L^2
(d) TV-L^2
(e) TVS-L^1+$H^1\mathbf{n}$-L^2
(f)

Figure 4.2: The comparison of different regularization in (2.5): the red region in (a) is an inpainting domain and (b) is the original image. In (c), the proposed model shows a proper reconstruction for both edges and ridges or valleys. Note that the image in (f) is reconstructed by $H^1\mathbf{n}$-L^2 with the exact normal vector field $\mathbf{n} = \nabla I$ of the original image.

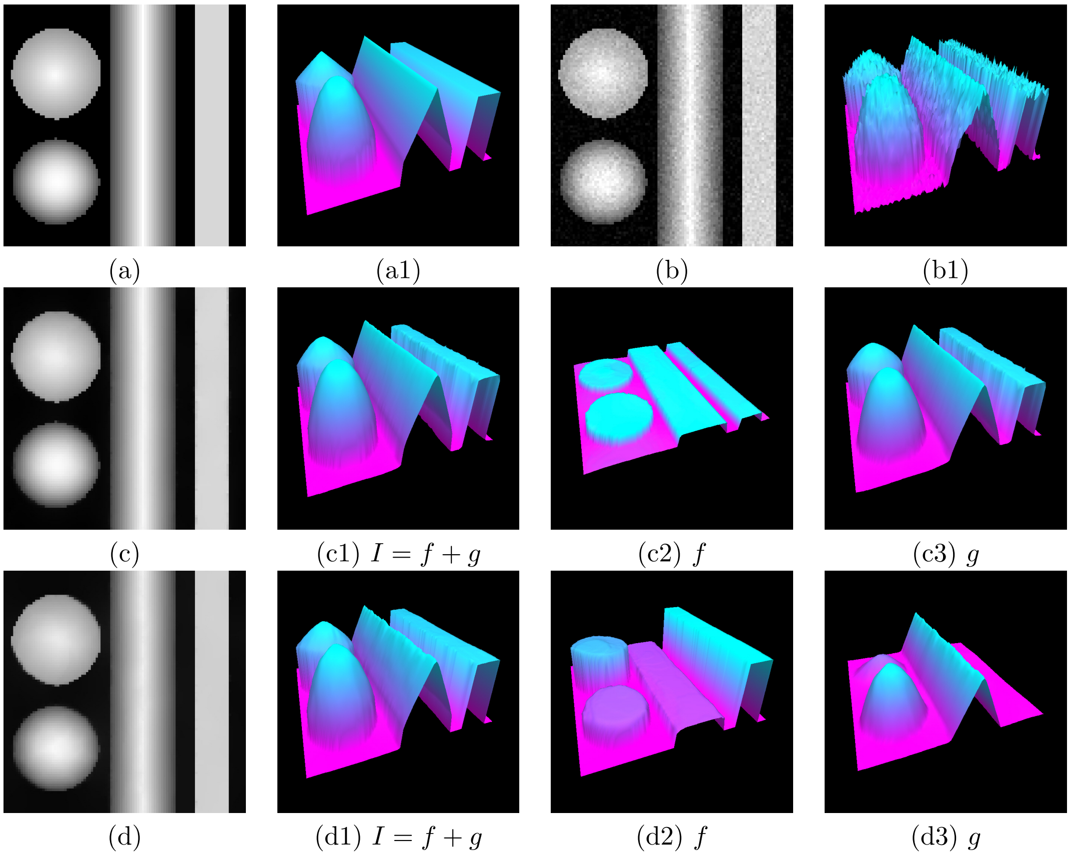


Figure 4.3: (a) is the original image and the Gaussian white noise with the standard deviation 10 is added in (b). (a1), (b1), (c1), and (d1) are the graph of the left image. (c1) is the sum of (c2) and (c3) which are the decomposition from the TVS-L^2+TV$\mathbf{n}$-L^2 model. (d1) is the sum of (d2) and (d3) which are the decomposition from the TVS-L^1+TV$\mathbf{n}$-L^2 model. In (d2) and (d3), we show image decomposition of an image into two parts: f in (2.15) which preserves the jump discontinuities and g in (2.14) which preserves the discontinuities of the image gradient or smooth regions. However, g in (c3) still has the jump discontinuities.

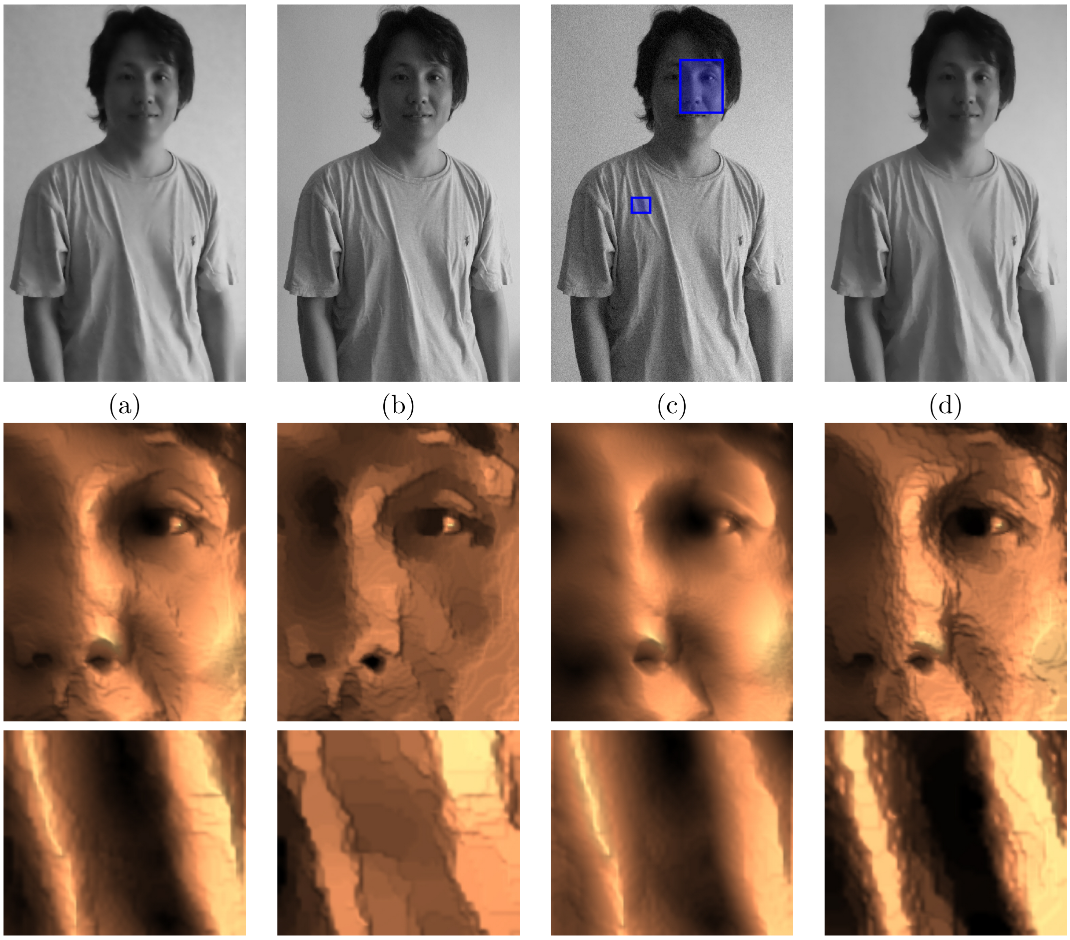
First of all, we observe that the L^1 and L^2 fidelity in the TVS-L^p model generates two distinctive regularized vector fields. In order to understand the different features, we need to observe the difference between L^1 and L^2 fidelity in TV-L^p model for image denoising [12, 13]. The TV-L^1 model outperforms the TV-L^2 model to remove the salt-and-pepper noise because the noise is considered as a large variation. The TV-L^2 model is suitable for eliminating a small variation of an image. Likewise, the TVS-L^1 model easily suppresses the salt-and-pepper type noise which can be considered as a large variation in a given vector data $\mathbf{t}^* = \nabla^\perp I^*$. If we have a piecewise constant image I^*, such a large variation in $\mathbf{t}^*$ happens at the jump discontinuities in the image. That is, the TVS-L^1 model easily suppresses the derivatives of jump discontinuities because they are considered as outliers in a given vector data $\mathbf{t}^* = \nabla^\perp I^*$. The TVS-$L^2$ model effectively works for denoising a small variation of the tangential vector field but it may not suppresses the derivative of jump discontinuities. Two distinctive regularized vector fields from the L^1 and L^2 fidelity in the TVS-L^p model are clearly shown in Figures 4.3-(c3) and 4.3-(d3).

In Figure 4.3, we show examples of image decomposition in the TVS-L^p+TV$\mathbf{n}$-L^2 model as we explained in Section 2.3. The different decompositions are expected from L^1 and L^2 fidelity in the first step. Generally, image decomposition from the TVS-L^p+TV$\mathbf{n}$-L^2 model is obtained as follows. First of all, we regularize tangential vector fields $\mathbf{n}$ from the TVS-L^p model. The function g in (2.14) is obtained by the TV$\mathbf{n}$ model and then the function f in (2.15) is obtained by the ROF model. From a noisy image I^* shown in (b), we obtain two regularized vector fields: one is $\mathbf{n}_2$ from the TVS-L^2 model and the other is $\mathbf{n}_1$ from the TVS-L^1 model. The surfaces in (c3) and (d3) are reconstructed by the TV$\mathbf{n}_2$ model and the TV$\mathbf{n}_1$ model, respectively. Since the TVS-L^1 model easily suppresses the derivatives of jump discontinuities in a given vector field, the surface (d3) does not have any jump discontinuities. However, the surface (c3) still has edges because of the L^2 fidelity. (c1) is the sum of f in (c2) and g in (c3) which are the decomposition from the TVS-L^2+TV$\mathbf{n}$-L^2 model. (d1) is the sum of f in (d2) and g in (d3) which are the decomposition from the TVS-L^1+TV$\mathbf{n}$-L^2 model. Even the final results in (c) and (d) are well recovered in the sense of image denoising, (c1) and (c2) are a correct decomposition because the image (c2) does not have jump discontinuities. Note that the function g in (2.14), shown in (c3) and (d3), is numerically obtained by the TV$\mathbf{n}$ model with $c_r = 1$ and $\epsilon_2 = 10^{-2}$. In (c), we use $c_r = 1$, $c_d = 1$, $\eta = 10$, and $\epsilon_1 = 10^{-3}$ for the TVS-L^2. In (c2), we use $c_r = 1$, $\xi = 20$, and $\epsilon_2 = 10^{-3}$ for finding a minimizer in (2.15). In (d), we use $c_r = 1$, $c_d = 1$, $c_f = 1$, $\eta = 1$, and $\epsilon_1 = 10^{-3}$ for the TVS-L^1. In (d2), we use $c_r = 1$, $\xi = 5$, and $\epsilon_2 = 10^{-3}$ for finding a minimizer in (2.15).

In Figure 4.4, a real example of image decomposition is shown. (a) is not a direct result of the TVS-L^1+TV$\mathbf{n}$-L^2 model in image denoising but the sum in (2.16) of two parts which are decomposed by the model. (d) is the result of the TV-L^2 model. In (d1), two blue regions in the result are magnified to show where sharp ridges or valleys are not preserved and stair-case effect happens. In (a2) and (a3), we show the results of image decomposition from the TVS-L^1+TV$\mathbf{n}$-L^2 model. Jump discontinuities and discontinuities of the image gradient or smooth regions are well decomposed and preserved. In (d), we use $c_r = 1$, $\xi = 15$, and $\epsilon_2 = 10^{-3}$. In (a), we use the TVS-L^1 model to obtain the regularized vector field $\mathbf{n}$ with $c_r = 1$, $c_d = 1$, $c_f = 1$, $\eta = 1$, and $\epsilon_1 = 10^{-3}$ and then we use the TV$\mathbf{n}$ model to obtain g with $c_r = 1$ and 10^{-2}. In order to obtain f in (2.15), we use $c_r = 1$, $\xi = 10^2$, and $\epsilon_2 = 10^{-2}$.

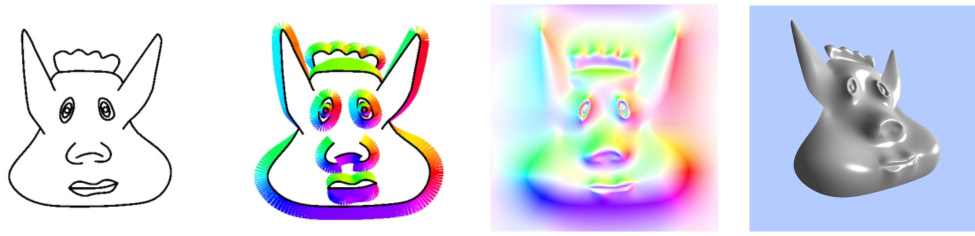
4.3 Surface reconstruction from sparse gradient

We present a reconstruction method of height function $z = I(x, y)$ on $\Omega \subset \mathbf{R}^2$ from a piecewise smooth gradient vector field on sparsely located curves $\Gamma \subsetneq \Omega$. The method called by surface reconstruction from sparse gradient is introduced in single view modeling [67–71]. From a given vector field $\mathbf{n}^*$ on sparsely located curves Γ in the domain Ω, an interpolated normal vector $\mathbf{n}$ is obtained over the whole domain by a PDE-based propagation [66] or a minimization method [68]. After that, surface-from-gradient methods [59, 80] are used to reconstruct a surface from the vector field $\mathbf{n}$.



(a1) $I = f + g$ (a2) f (a3) g (d1) TV-L^2

Figure 4.4: (a) is not a direct result of the TVS-L^1+TV$\mathbf{n}$-L^2 model in image denoising but the sum in (2.16) of two parts which are decomposed by the model. (b) is the original image and the Gaussian white noise with the standard deviation 10 is added in (c). (d) is the results of the TV-L^2 model. Blue regions in (c) are magnified in the second row and the third row. They are shown by a surface rendering from the top view. (a2) and (a3) are the decomposition from the TVS-L^1+TV$\mathbf{n}$-L^2 model. In (a2) and (a3), f in (2.15) preserves the jump discontinuities and g in (2.14) preserves discontinuities of the image gradient or smooth regions.



(a) (b) (c) (d)

Figure 4.5: The procedure to reconstruct a surface from sparse gradient: (a) is a basic drawings to indicate curves Γ. (b) is an assigned vector field $\mathbf{n}^*$ on Γ. (c) is an inpainted vector field $\mathbf{n}$ by the H^1S-L^1 model rendered by the HSV color map. (d) is a reconstructed surface by the TV$\mathbf{n}$ model.

In this subsection, we apply the proposed models (2.4) and (2.5) to surface reconstruction from sparse gradient. The procedure is illustrated in Figure 4.5. From a given vector field $\mathbf{n}^*$ in (b) on curves Γ in (a), we use the following models to obtain an inpainted vector field on Ω:

$$\text{TVS-}L^1 \quad \text{or} \quad H^1\text{S-}L^1, \tag{4.3}$$

where $(\mathbf{t}^*)^\perp = \mathbf{n}^*$. For instance, (c) is an inpainted vector field by the H^1S-L^1 model rendered by the HSV color map. After the inpainted vector field is obtained, we use the TV$\mathbf{n}$ model to reconstruct a height function in (d). In this case, a height value is arbitrarily fixed at a point because a minimizer in the TV$\mathbf{n}$ model is obtained up to constants. If there is a prior information of height, the TV$\mathbf{n}$-L^p ($p \geq 1$) model also can be used. Since the divergence free constraint in (4.3) is crucial to inpaint tangential vectors over large regions as we observed in Figure 4.1, it is not necessary to consider VTV-L^1 and VH^1-L^1 model in this application. Note that the corresponding constraint to divergence free condition in the normal vectors is called as the integrability condition. It has been used in many literatures of surface reconstruction; see [59–62] and references therein.

We initially set up vectors $\mathbf{n}^*$ on given sparse curves Γ as perpendicular to the curves. A default magnitude of vectors $\mathbf{n}^*$ is a constant value on all curves. Since we assume the assigned vectors $\mathbf{n}^*$ as a projection of 3D surface normal vectors on the domain Ω, the height of reconstructed surface is decreasing along the direction of vector and larger magnitude of vectors reconstruct steeper surface. So, we can adjust the vectors $\mathbf{n}^*$ on Γ in order to reconstruct a desired surface.

The main focus in this subsection is to observe effects of TV and H^1 regularization in (4.3) in terms of surface reconstruction. As we have observed in Section 4.1, the TV regularization preserves discontinuities in the inpainted vector field and the H^1 regularization smears them out. It means that the TV regularization reconstructs sharp ridges or valleys from discontinuities in the given vector field $\mathbf{n}^*$ on Γ and the H^1 regularization can not recover sharp ridges or valleys. In Figure 4.6, we illustrate such a difference. (a1) and (b1) are initial drawings to indicate Γ in (2.4) and (a2) and (b2) are the initial vector field $\mathbf{t}^*$ on Γ presented by the HSV color map. Discontinuities in the vector field $\mathbf{t}^*$ on Γ are easily seen because of sudden color changes in the HSV color map. There are four discontinuous points at the corners of the outer square. In (a2), there is another discontinuous point located at the junction of the cross shape. In (b2), there are three more discontinuous points on the cross shape. In (c) and (d), we observe that these discontinuities are well propagated into the domain under the TV regularization. However, these discontinuities in (e) and (f) are smeared out because of the H^1 regularization. In Figure 4.6, we fix the error bounds $\epsilon_1 = \epsilon_2 = 10^{-3}$ and we use $c_r = 10^{-2}$, $c_f = 1$, and $\epsilon_2 = 10^{-3}$ for the TV$\mathbf{n}$ model. In (c) and (d), we use $c_r = 10$, $c_d = 1$, $c_f = 1$, and $\eta = 100$ for the TVS-L^1 model. In (e) and (f), we use $c_d = 1$, $c_f = 1$, and $\eta = 100$.

In Figure 4.7, we use an example drawn by an artist. (a) is a basic sketch of curves and (b) is the assigned vector field on the curves. (c1) and (d1) are the results from TVS-L^1 model and H^1S-L^1 model, respectively, presented by the HSV color map. After we obtain inpainted vector

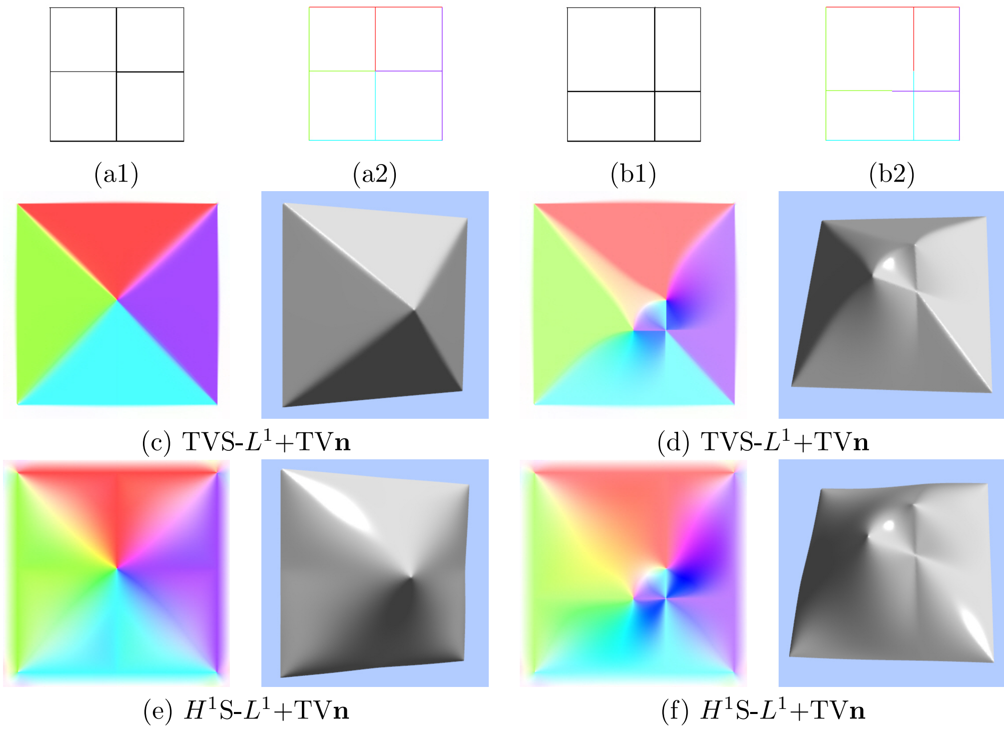


Figure 4.6: The comparison between TVS-L^1 model and H^1S-L^1 model: (a1) and (b1) are initial drawings to indicate Γ in (2.4) and (a2) and (b2) are the initial vector field $\mathbf{t}^*$ on Γ presented by the HSV color map. The image on the left of each surface is the HSV color map of result from the inpainted vector field. In (c) and (d), since the TV norm preserves discontinuities in the vector field, the crease structure is well propagated from the singular points in a given vector field on Γ. The crease structure is smeared out under the H^1 norm shown in (e) and (f).

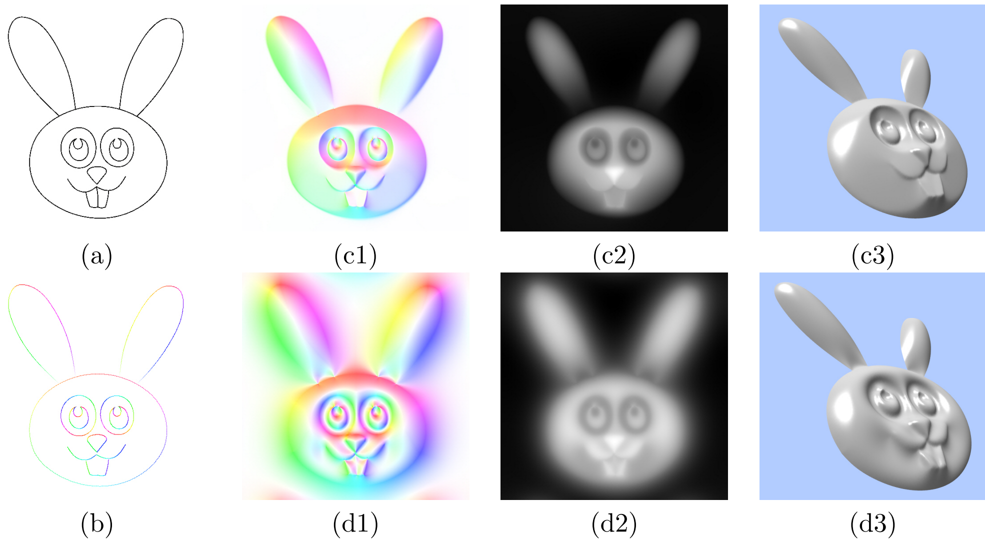


Figure 4.7: (a) is the curves Γ drawn by an artist. (b) is the assigned vector field $\mathbf{t}^*$ on the curves and (c1) and (d1) are the result from TVS-L^1 model and H^1S-L^1 model, respectively, presented by the HSV color map. (c2) and (d2) are the scaled height map obtained from (c1) and (d1), respectively, via the TV$\mathbf{n}$ model. (c3) and (d3) are reconstructed surfaces. The different shapes between (c3) and (d3) are caused by the different norms.

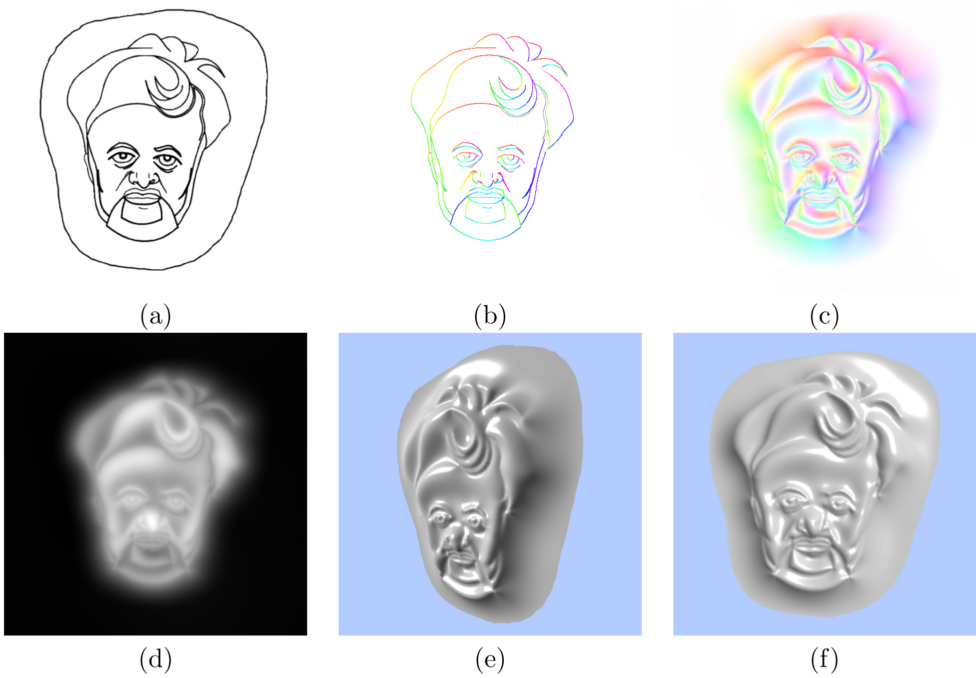
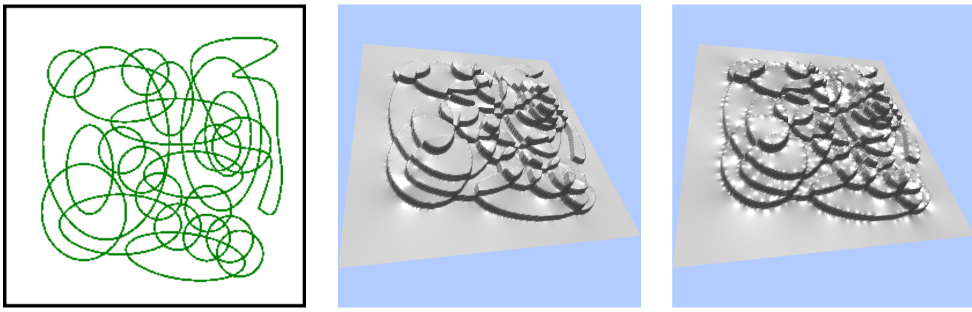


Figure 4.8: (a) is the curves Γ drawn by an artist. (b) is the assigned vector field $\mathbf{t}^*$ on the curves and (c) is the result from the H^1S-L^1 model presented by the HSV color map. (d) is the scaled height map obtained by the TV$\mathbf{n}$ model. (e) and (f) are two views of the reconstructed surface.



(a) (b) TV$\mathbf{n}$ (c) $H^1\mathbf{n}$

Figure 4.9: The comparison between TV$\mathbf{n}$ model and $H^1\mathbf{n}$ model from the dense normal vector field: (a) shows how the vector field $\mathbf{n}$ is defined. $\mathbf{n} = 0$ is used except on the green curves. The magnitude of $\mathbf{n}$ is a constant on the curves and the direction is the outward normal to the curves. The result in the $H^1\mathbf{n}$ model has oscillatory artifacts in the height field along the curves but the TV$\mathbf{n}$ model reconstructs jump discontinuities.

fields $\mathbf{n}$ in (c1) and (d1), the TV$\mathbf{n}$ model reconstructs surfaces in (c3) and (d3), respectively. They are clearly different shapes because of different norms in the regularization term. In Figure 4.8, more complicated curves are used. In this case, it seems to be difficult to initially assign a proper vector field on given curves in order to reconstruct a desired surface. The authors [68] used a 3D reference model to intuitively assign suitable vectors on the curves. In this paper, we simply use the perpendicular direction to the given curves. We also modulate the magnitude of vectors to change the steepness of surface. The magnitude of some vectors at the T junction is gradually decreased to reduce artifacts in the surface. In Figures 4.7 and 4.8, we use the same parameters for the TVS-L^1+TV$\mathbf{n}$ model and the H^1S-L^1+TV$\mathbf{n}$ model as in Figure 4.6.

The TV$\mathbf{n}$ model plays an important role to reconstruct jump discontinuities in a height map if the vector field $\mathbf{n}$ has the discrete derivative information of the jump. The $H^1\mathbf{n}$ model can not recover jump discontinuities because the minimizer should be in the H^1 space. In Figure 4.9, we make an extreme case that the vector field $\mathbf{n} = 0$ is used except on the green curves in (a), the magnitude of $\mathbf{n}$ is a constant on the curves, and the direction is the outward normal to the curves. The result in the $H^1\mathbf{n}$ model has oscillatory artifacts in the height field along the curves but the TV$\mathbf{n}$ model reconstructs jump discontinuities. Usually, these artifacts have been partially removed by approximating the locations of jump discontinuities. The approximate positions are used as a weight function to reduce overshooting or undershooting artifacts. The issue in Figure 4.9 is that such an extra approximation is certainly unnecessary if we use the TV$\mathbf{n}$ model for surface reconstruction.

4.4 Direction denoising

Direction denoising is an application of regularizing a given noisy vector field. That is, we do not need to use the second step (2.5) in direction denoising. Since a general vector field does not satisfy the divergence free constraint, we do not consider the constraint in this application. Even though direction denoising is not related to two-step methods, the main reason we present this application is to compare effect of L^1 and L^2 fidelity in the first step (2.4). For this purpose, we use the following two models in (2.4):

$$\text{VTV-}L^p \quad \text{and} \quad \text{V}H^1\text{-}L^p, \tag{4.4}$$

where $p = 1$ or 2.

In order to understand the different features between L^1 and L^2 fidelity in (4.4), we need to observe the difference between TV-L^1 and TV-L^2 model for image denoising [10]. The L^2 fidelity in TV-L^2 model makes contrast reduction effect in image denoising [9]. However, the L^1 fidelity is shown to preserve the contrast of an image [10–13]. Likewise, we would like to observe the contrast reduction effect in a denoised vector field with the L^2 fidelity in (4.4). Also, we numerically observe whether the L^1 fidelity prevents the contrast reduction effect or not.

The contrast reduction in direction denoising is not only the reduction of magnitude of vectors but also the reduction of direction; see Figure 4.10. (a) shows the HSV color map of an original unit vector field. The size of domain is 128×128. In (c) and (d), the Gaussian white noise with a standard deviation 30 and the salt-and-pepper noise with a noise density 0.3 are added into the angles of vectors in (a), respectively. Note that the salt-and-pepper noise of the angles of vectors is considered as flipping the direction of the vectors oppositely. In order to observe the quality of a regularized vector field $\mathbf{t}$ from different models in (4.4), we compute the integral curves:

$$\frac{d}{d\tau}\mathbf{x}(\tau) = \mathbf{t}\left(\mathbf{x}(\tau)\right), \tag{4.5}$$

where the initial points $\mathbf{x}(0) = (16, 68)$, $(32, 68)$, and $(64, 68)$ are used and they are shown in red, blue, and green curves, respectively. The black dotted integral curves in (b) are obtained in the original vector field. The above equation is solved by the algorithm `stream2` in the MATLAB with the time step 10^{-4}. The images in the second row and the fourth row are the regularized vector fields from different models in (4.4) rendered by the HSV color map. In the third row and the fifth row, we show the integral curves in the above denoised vector field.

In Figures 4.10-(c3), 4.10-(c4), 4.10-(d3), and 4.10-(d4), we use VTV-L^2 or VH^1-L^2 models to denoise given vector fields in (c) and (d). The L^2 fidelity causes the reduction of magnitude and direction in denoised vectors. The magnitude reduction is shown as changing of saturation in the HSV color map on the second column and the fourth column. The direction reduction is shown in the corresponding integral curves. That is, the slope of all curves in (c3), (c4), (d3), and (d4) is reduced starting from the initial points of integral curves. We show the results from VTV-L^1 or VH^1-L^1 models in (c1), (c2), (d1), and (d2). Compared to the L^2 fidelity, the L^1 fidelity shows better performance on preserving magnitude and direction in denoised vectors. In Figure 4.10, we use error bound $\epsilon_1 = 10^{-3}$. In (c1), $c_r = 10$, $c_f = 0.4$, and $\eta = 0.7$ are used. In (c2), $c_f = 1$ and $\eta = 0.1$ are used. In (c3), $c_r = 1$ and $\eta = 0.7$ are used. In (c4), $\eta = 5 \cdot 10^{-2}$ is used. In (d1), $c_r = 1$, $c_f = 1$, and $\eta = 0.8$ are used. In (d2), $c_f = 1$ and $\eta = 0.5$ are used. used. In (d3), $c_r = 10^2$ and $\eta = 0.8$ are used. In (d4), $\eta = 0.1$ is used.

The VTV-L^1 model is used to denoise of a real diffusion tensor imaging data in Figure 4.11. In the second row, the major stream is preserved while the noisy directions are rearranged into the stream. In this example, $c_r = 10^2$, $c_f = 1$, $\eta = 1.5$, and $\epsilon_1 = 10^{-3}$ are used.

If each component of $\mathbf{t}$ in (2.4) presents each RGB channel of a color image $I : \Omega \to \mathbf{R}^3$, the proposed general form (2.4) with $F = 0$ is the same as the vectorial TV-L^p model ($p \geq 1$) which has been shown as a model to denoise a color image; see [5] and references therein. In Figures 4.12 and 4.13, we show the examples of using the L^1 fidelity to denoise the salt-and-pepper noise with a noise density 0.5. As we have observed that the L^1 fidelity is better than the L^2 fidelity in terms of a contrast preservation in direction denoising in Figure 4.10, the RGB color vectors in Figures 4.12-(c) and 4.13-(c) are well recovered as the original colors. The images in Figures 4.12-(d) and 4.13-(d) are obtained by the command `medfilt2(I,[3 3])` in the MATLAB. In Figure 4.12-(c), we use $c_r = 10$, $c_f = 0.2$, $\xi = 0.7$, and $\epsilon_2 = 10^{-2}$. In Figure 4.13-(c), we use $c_r = 1$, $c_f = 0.4$, $\xi = 0.7$, and $\epsilon_2 = 10^{-2}$.

4.5 Image denoising

The two-step methods for image denoising in [56, 57, 63] recover jump discontinuities of image and discontinuities of image gradient while reducing stair-case effect. In this subsection, we show

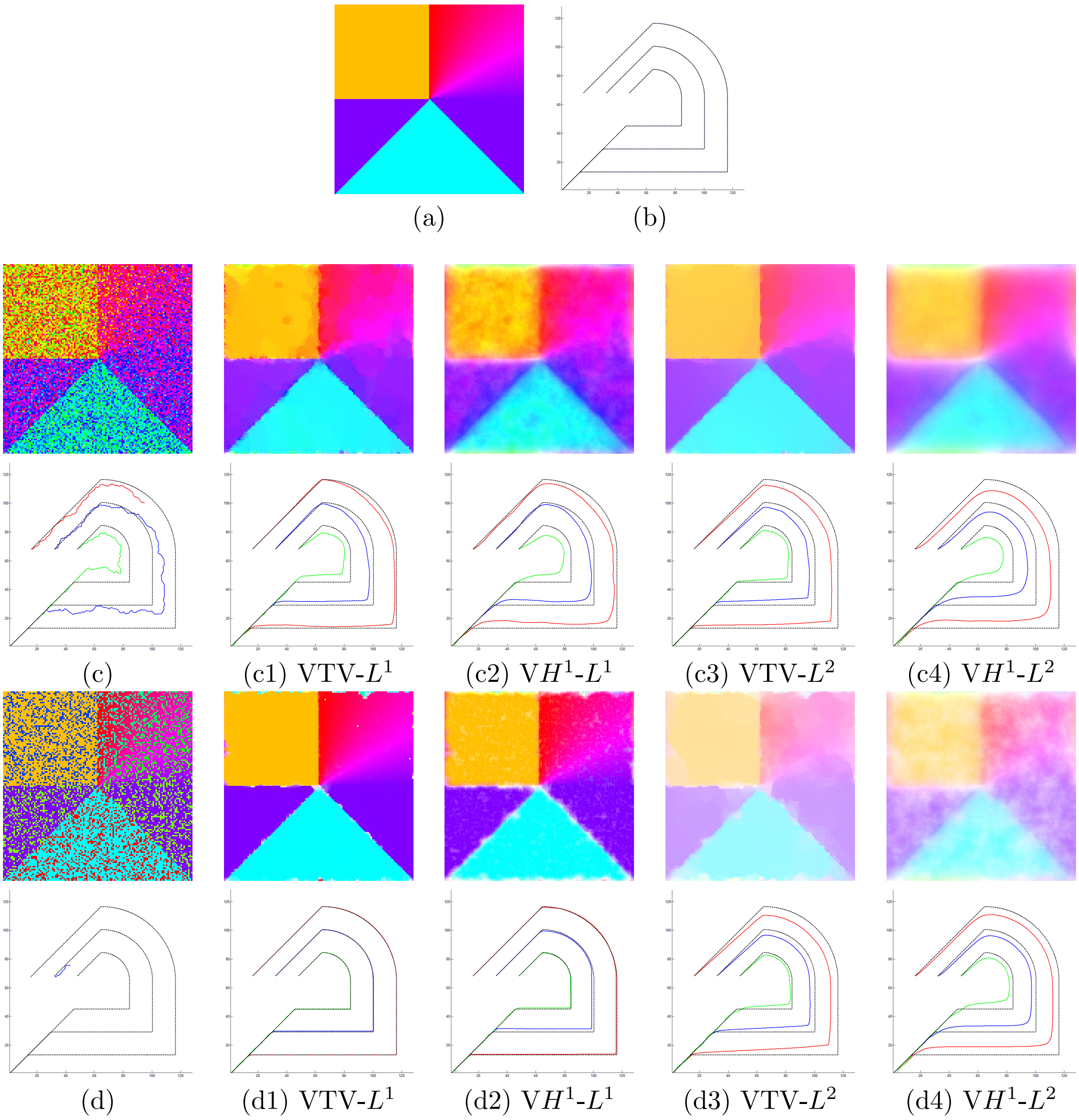


Figure 4.10: (a) shows the HSV color map of the original unit vector field. The black dotted curves are obtained by (4.5) in the original vector field. The other curves are obtained in the regularized vector field. In (c) and (d), the Gaussian white noise with a standard deviation 30 and the salt-and-pepper noise with a noise density 0.3 are added into the angles of vectors in (a), respectively. The TV regularization preserves sudden directional changes. The L^1 fidelity makes smaller amount of contrast reduction in the vector field than the L^2 fidelity.

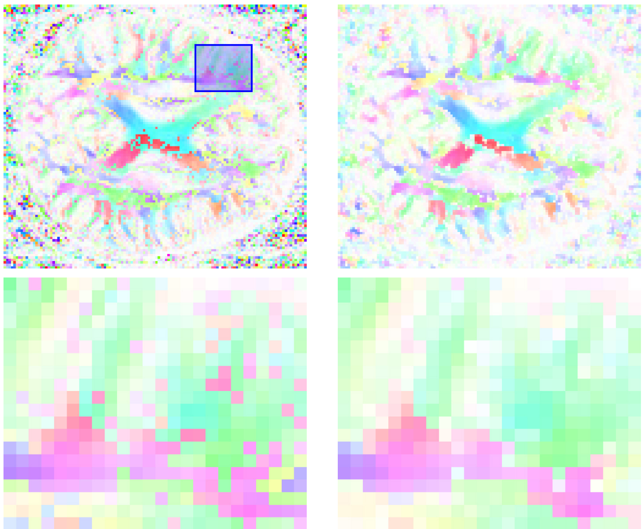
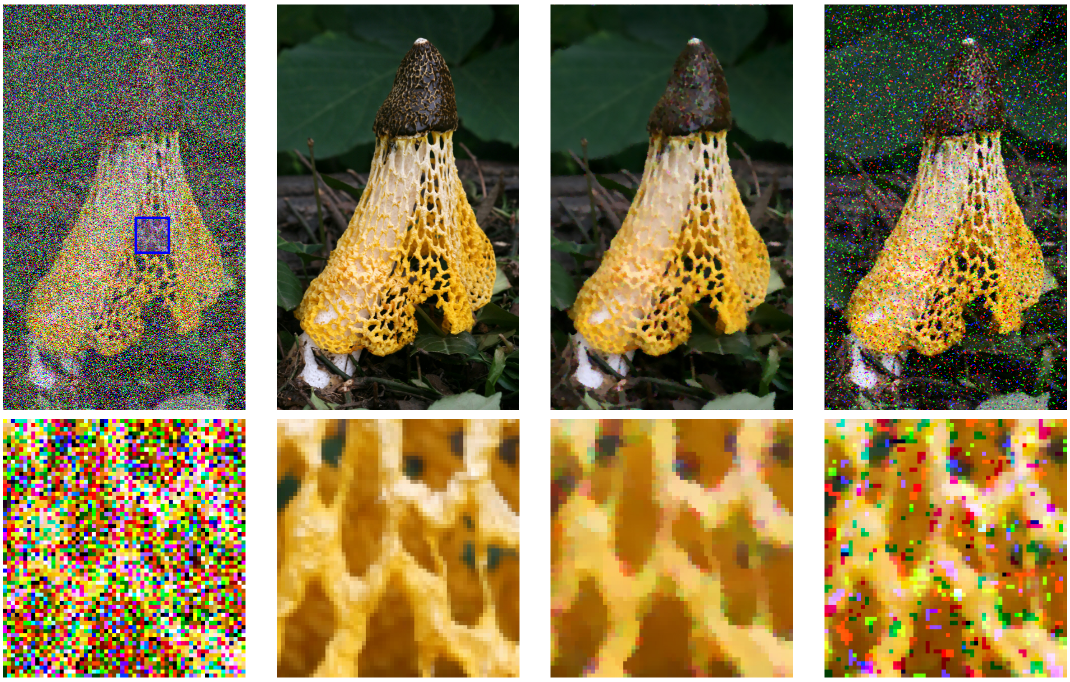


Figure 4.11: The result of the VTV-L^1 model for denoising direction of a real DTI data: the left images are a given data and the right images are the result of the VTV-L^1 model. Vectors are rendered by the HSV color map. The second row is the part on a small blue region. The major stream is preserved while the noisy directions are rearranged into the steam.

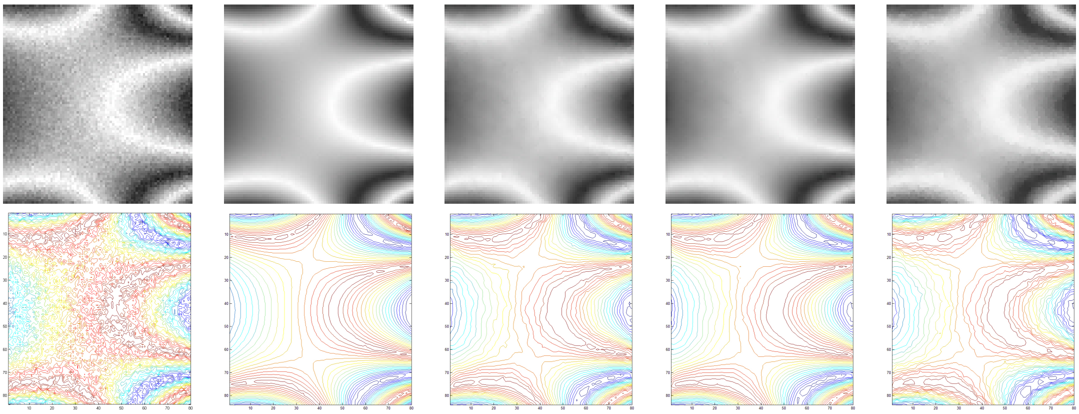
(a) (b) (c) (d)

Figure 4.12: In (a), the salt-and-pepper noise with a noise density 0.5 is added in the synthetic image (b). (c) is the result of vectorial color TV model with the L^1 fidelity. (d) is the result of a median filter in each RGB channel. As the VTV-L^1 model preserves the orientation and the magnitude of vectors in Figure 4.10, the result in (c) preserves the original color very well.



(a) (b) (c) (d)

Figure 4.13: In (a), the salt-and-pepper noise with a noise density 0.5 is added in the real image (b). (c) is the result of vectorial color TV model with the L^1 fidelity. (d) is the result of a median filter in each RGB channel. The VTV-L^1 model preserves the color very well.



(a) (b) (c) SNR: 26.72 (d) SNR: 26.44 (e) SNR: 23.49

Figure 4.14: In (a), we add the Gaussian white noise with a standard deviation 10 in the synthetic original image (b). (c), (d), and (e) are results of the TVS-L^1+TV**n**-L^2 model, the TVS-L^2+TV**n**-L^2, and the TV-L^2 model, respectively. The second row is a isocontour plot of the corresponding images in the first row. Stair-case effect is easily observed in the TV-L^2 model. However, the TV-Stokes equation reconstructs smooth structures as well as sharp changes in the image gradient.

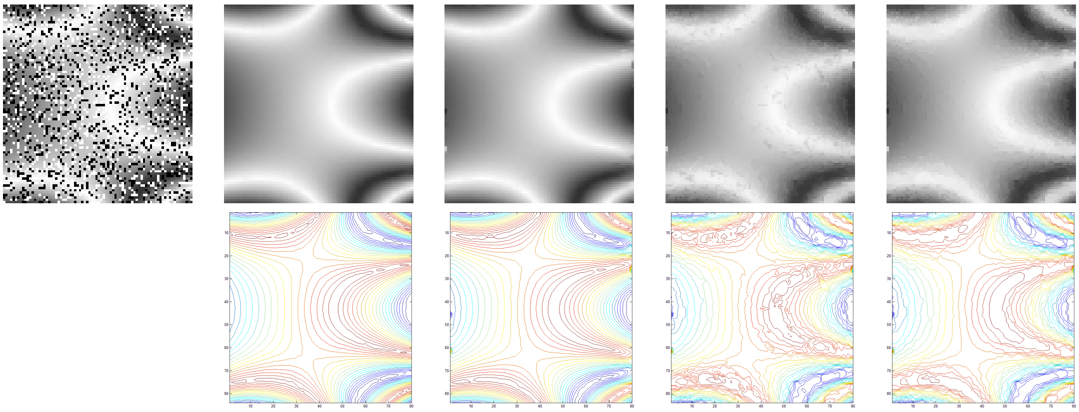
these properties in the proposed general TV-Stokes model:

$$\text{TVS-}L^{p_1} + \text{TV}\mathbf{n}\text{-}L^{p_2}. \tag{4.6}$$

Let us discuss about the choice of p_1 and p_2 depending on types of noise. In the second step TV**n**-L^{p_2}, it is reasonable to use $p_2 = 1$ for the salt-and-pepper noise and $p_2 = 2$ for the Gaussian white noise. We suggest to use $p_1 = 1$ in the first step TVS-L^{p_1}. In the presence of the outliers in an image, its discrete derivatives $\mathbf{t}^*$ also have outliers and then $p_1 = 1$ should be used. The L^2 fidelity is efficient to denoise a small variation of data, but it makes a contrast reduction in general. That is, the use of $p_1 = 2$ in the first step changes the magnitude or direction of tangential vectors and then it affects the reduction of tangent in the image. Note that the magnitude reduction of regularized vectors is also observed in Section 4.4. Even though the effect of reduction in a regularized vector field is not substantial in image denoising, it is desirable to use $p_1 = 1$ because of the compatibility of norms between the fidelity in the TVS-L^{p_1} model and the regularization in the TV**n**-L^{p_2} model; see more details in Section 2.1. The choice of $p_1 = 1$ will be verified in the examples shown in Figures 4.14 and 4.15.

In Figure 4.14, we show an effect of L^1 and L^2 fidelity in TVS-L^{p_1} model under the Gaussian white noise. For fair comparison, the TV**n**-L^2 model is fixed in the second step. In (a), we add the Gaussian white noise with a standard deviation 10 in the synthetic original image (b). (c) and (d) are results of TVS-L^1+TV**n**-L^2 model and the TVS-L^2+TV**n**-L^2, respectively. Even though the effect of contrast reduction in the regularized vector field is caused by the L^2 norm in the first step, the reduction of tangent in a denoised image is not severely noticeable in (c) and (d). In (e), stair-case effect is easily observed in the TV-L^2 model. From the isocontour plot in the result of the TVS-L^p+TV**n**-L^2 ($p = 1$ or 2), one can see that the TV-Stokes equation reconstructs smooth structures as well as sharp changes in the image gradient. In (c), we use $c_r = 10^2$, $c_d = 1$, $c_f = 1$, $\eta = 1$, and $\epsilon_1 = 10^{-2}$ for the TVS-L^1 model and $c_r = 1$, $\xi = 28$, and $\epsilon_2 = 10^{-2}$ for the TV**n**-L^2 model. In (d), we use $c_r = 1$, $c_d = 1$, $\eta = 18$, and $\epsilon_1 = 10^{-2}$ for the TVS-L^2 model and $c_r = 1$, $\xi = 18$, and $\epsilon_2 = 10^{-2}$ for the TV**n**-L^2 model. In (e), $c_r = 1$, $\xi = 25$, and $\epsilon_2 = 10^{-2}$ are used for the TV-L^2 model.

In Figure 4.15, we show an effect of L^1 and L^2 fidelity in TVS-L^{p_1} model under the salt-and-



(a) (b) (c) SNR: 23.66 (d) SNR: 18.51 (e) SNR: 19.21

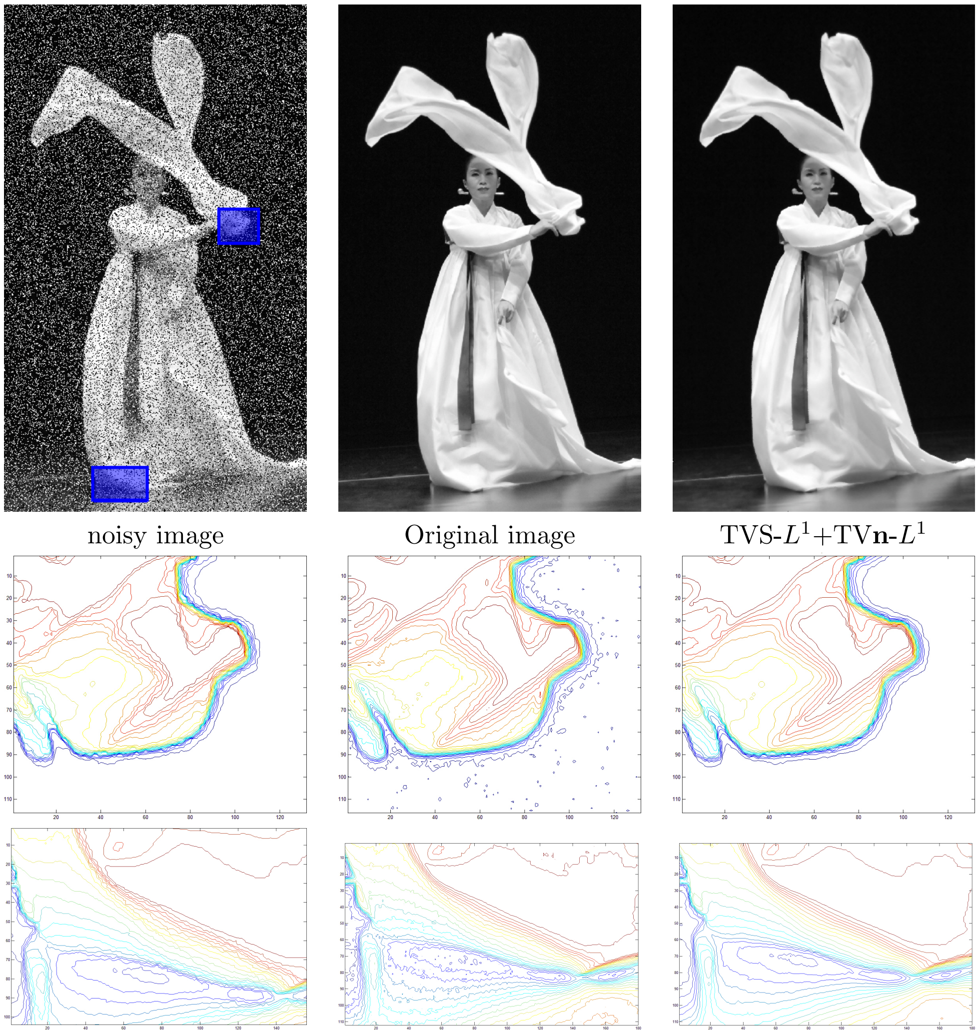
Figure 4.15: In (a), we add the salt-and-pepper noise with a noise density 0.3 in the synthetic original image (b). (c), (d), and (e) are the results of the TVS-L^1+TV$\mathbf{n}$-L^1 model, the TVS-L^2+TV$\mathbf{n}$-L^1 model, and the TV-L^1 model, respectively. The second row is a isocontour plot of the corresponding images in the first row. The TV-Stokes equation reconstructs smooth structures as well as sharp changes in the image gradient.

pepper noise. For fair comparison, the TV$\mathbf{n}$-L^2 model is fixed in the second step. In (a), we add the salt-and-pepper noise with a noise density 0.3 in the original image (b). (c) and (d) are results of TVS-L^1+TV$\mathbf{n}$-L^1 model and the TVS-L^2+TV$\mathbf{n}$-L^1, respectively. Since the TVS-L^2 model does not effectively surpass the outliers in the noisy tangential vector field, the result of the TVS-L^2+TV$\mathbf{n}$-L^1 model still has some noisy features in (d). However, the TVS-L^1 model in the first step efficiently eliminates the salt-and-pepper noise in a regularized tangential vector field. In (c), combining with the TV$\mathbf{n}$-L^1 model in the second step, discontinuities of the image gradient and jump discontinuities of image are well preserved while reducing stair-case effect. In (e), the TV-L^1 model is also able to denoise the salt-and-pepper noise, but it does not preserve discontinuities of the image gradient and stair-case effect is observed. In (c), we use $c_r = 1$, $c_d = 1$, $c_f = 1$, $\eta = 0.8$, and $\epsilon_1 = 10^{-2}$ for the TVS-L^1 model and $c_r = 1$, $c_f = 1$, $\xi = 0.8$, and $\epsilon_2 = 10^{-2}$ for the TV$\mathbf{n}$-L^1 model. In (d), we use $c_r = 10^2$, $c_d = 1$, $\eta = 2.6$, and $\epsilon_1 = 10^{-2}$ for the TVS-L^2 model and $c_r = 1$, $\xi = 1$, and $\epsilon_2 = 10^{-2}$ for the TV$\mathbf{n}$-L^2 model. In (e), $c_r = 1$, $c_f = 1$, $\xi = 1$, and $\epsilon_2 = 10^{-2}$ are used for the TV-L^1 model.

In Figure 4.16, we use the TVS-L^1+TV$\mathbf{n}$-L^1 model to remove the salt-and-pepper noise with a noise density 0.3 in a real image. The second row and the third row are the contour plots in the blue regions. (a), (b), and (c) are the contour plot from the result of TV-L^1, original image, and the result of TVS-L^1+TV$\mathbf{n}$-L^1, respectively. From the comparison with the TV-L^1 model in the contour plots, we observe that the TVS-L^1+TV$\mathbf{n}$-L^1 model preserves jump discontinuities, smooth regions, and discontinuities of the image gradient. We use $c_r = 10$, $c_d = 1$, $c_f = 1$, $\eta = 0.8$, and $\epsilon_1 = 10^{-2}$ in the first step and $c_r = 1$, $c_f = 1$, $\xi = 0.8$, and $\epsilon_2 = 10^{-2}$ in the second step. For the TV-L^1 model, we use $c_r = 1$, $c_f = 1$, $\xi = 0.8$, and $\epsilon_2 = 10^{-2}$.

5 Conclusion

In this paper, we generalize the basic TV-Stokes model (2.2) and (2.3), provide an efficient and fast numerical algorithm based on the augmented Lagrangian method, and show that the proposed general form can be used for a number of applications. We generalize the first step in [45, 58] and the second step in [63]. The generalized form uses TV or H^1 regularization, the L^p norm ($p \geq 1$) in the fidelity term, and the arbitrary integration domain in the fidelity term. The use of



(a) TV-L^1
SNR: 30.60

(b) Original

(c) TVS-L^1+TV$\mathbf{n}$-L^1
SNR: 31.54

Figure 4.16: We use the TVS-L^1+TV$\mathbf{n}$-L^1 model to remove the salt-and-pepper noise with a noise density 0.3 in the real image. From the left in the first row, noisy image, original image, and the result of TVS-L^1+TV$\mathbf{n}$-L^1 are illustrated. The second row and the third row are the contour plots in the blue regions. (a), (b), and (c) are the contour plot from the result of TV-L^1, original image, and the result of TVS-L^1+TV$\mathbf{n}$-L^1, respectively. The contour plot shows that the TVS-L^1+TV$\mathbf{n}$-L^1 model clearly eliminates the salt-and-pepper noise and preserve discontinuities of the image gradient while reducing stair-case effect.

arbitrary integration domain makes it possible to reconstruct a surface from sparse gradient. We illuminate a property of the generalized TV-Stokes model which is to decompose an image into jump discontinuities of a data and discontinuities of the data gradient or smooth regions. The proposed model can be applied to many applications such as image inpainting, image decomposition, surface reconstruction from sparse gradient, direction denoising, and image denoising. We investigate the various effects from using different norms via many examples in applications. Numerical experiments demonstrate that the proposed model recovers jump discontinuities of a data and discontinuities of the data gradient while reducing stair-case effect.

Acknowledgement

We would like to thank Prof. Hock Soon Seah, Dr. Jie Qiu, Dr. Eiji Sujisaki, Lei Jia, and Cao Youfang in School of Computer Engineering, Nanyang Technological University, Singapore for providing the initial drawings, setting the initial vectors, and rendering the surfaces in Figures 4.6, 4.7, and 4.8. Prof. Yong Nam Lee owns the copyright of Figure 4.16 which is shown in *http://blog.chosun.com/ynlee40/3449246*. The copyright of Figure 4.13 is owned by Mr. Byung Keun Hahn. We specially thank Seung Hee Lee who performed a Korean traditional dance in Figure 4.16 and the photographers to allow us to use beautiful images.

References

[1] F. Cole, A. Golovinskiy, A. Limpaecher, H. S. Barros, A. Finkelstein, T. Funkhouser, and S. Rusinkiewicz. Where do people draw lines? *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 27(3), 2008.

[2] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[3] R. P. Fedkiw, G. Sapiro, and C.-W. Shu. Shock capturing, level sets, and PDE based methods in computer vision and image processing: a review of osher's contributions. *J. Comput. Phys.*, 185:309–341, 2003.

[4] T. Chan, S. Esedoḡlu, F. Park, and A. Yip. Recent developments in total variation image restoration. In *Mathematical Models of Computer Vision*. Springer Verlag, 2005.

[5] X. Bresson and T. F. Chan. Fast daul minimization of the vectorial total variation norm and applications to color image processing. *Inverse Problems and Imaging*, 2:455–484, 2008.

[6] W. Ring. Structural properties of solutions to total variation regularization problems. *Math. Modleing Numer. Anal.*, 34:799–810, 2000.

[7] T.F. Chan, A. Marquina, and P. Mulet. High-order total variation-based image restoration. *SIAM J. Sci. Comput.*, 22:503–516, 2000.

[8] M. Nikolova. Local strong homogeneity of a regularized estimator. *SIAM J. Appl. Math.*, 61:633–658, 2007.

[9] D. Strong and T. F. Chan. Edge-preserving and scale-dependent properties of total variation regularization. *Inverse Problems*, 19:165–187, 2003.

[10] T. F. Chan and S. Esedoḡlu. Aspects of total variation regularized l^1 function approximation. *SIAM J. Appl. Math.*, 65:1817–1837, 2005.

[11] S. Alliney. Digital filters as absolute norm regularizers. *IEEE Trans. Sigal Process.*, 40:1548–1562, 1992.

[12] M. Nikolova. Minimizers of cost-functions involving non-smooth data-fidelity terms. *SIAM J. Numer. Anal.*, 40:965–994, 2002.

[13] M. Nikolova. A variational approach to remove outliers and impulse noise. *J. Math. Imag. Vis.*, 20:99–120, 2004.

[14] A. Chambolle and P.-L. Lions. Image recovery via total variation minimization and related problems. *Numer. Math.*, 76:167 – 188, 1997.

[15] T. Chan and L. Vese. Fourth-order partial differential equaion for noise removal. *IEEE Trans. Image Process.*, 9(10):1723–1730, 2000.

[16] M. Lysaker, A. Lundervold, and X.-C. Tai. Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time. *IEEE Trans. Image Process.*, 12(12):1579–1590, 2003.

[17] J. B. Greer and A. L. Bertozzi. Traveling wave solutions of fourth order PDEs for image processing. *SIAM J. Math. Anal.*, 36:38–68, 2004.

[18] M. Lysaker and X.-C. Tai. Iterative image restoration combining total variation minimization and a second-order functional. *Int. J. Comput. Vis.*, 66:5–18, 2006.

[19] F. Li, C. Shen, J. Fan, and C. Shen. Image restoration combining a total variational filter and a fourth-order filter. *J. Visul Comm. Image Rep.*, 18:322–330, 2007.

[20] S. Didas, J. Weickert, and B. Burgeth. Properties of higher order nonlinear diffusion filtering. *J. Math. Imaging Vis.*, 35:208–226, 2009.

[21] J. Ling and A. C. Bovik. Smoothing low-SNR molecular images via anisotropic median-diffusion. *IEEE Trans. Image Process.*, 21:377–384, 2007.

[22] J. Rajan, K. Kannan, and M. R. Kaimal. An improved hybrid model for molecular image denoising. *J. Math. Imaging Vis.*, 31:73–79, 2008.

[23] A. Buades, B. Coll, and J.-M. Morel. The staircasing effect in neighborhood filters and its solution. *IEEE Trans. Image Process.*, 15:1499–1505, 2006.

[24] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Machine Intell.*, 12(7):629–639, 1990.

[25] T.F. Chan, G.H. Golub, and P. Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM J. Sci. Comput.*, 20:1964 – 1977, 1999.

[26] A. Chambolle. An algorithm for total variational minimization and applications. *J. Math. Imaging Vis.*, 20:89 – 97, 2004.

[27] M. Hintermüller and G. Stadler. An infeasible primal-dual algorithm for total bounded variation-based inf-convolution-type image restoration. *SIAM J. Sci. Comput.*, 28(1):1–23, 2006.

[28] M. Zhu and T. Chan. An efficient primal-dual hybrid gradient algorithm for total variation image restoration. Technical report, UCLA CAM Report 08-34, 2008.

[29] X. Zhang, M. Burger, and S. Osher. A unified primal-dual algorithm framework based on Bregman iteration. Technical report, UCLA CAM Report 09-99, 2009.

[30] Y. Dong and M. Hintermüller anf M. Neri. An efficient primal-dual method for L^1-TV image restoration. *SIAM J. Imaging Science*, 2(4):1168–1189, 2009.

[31] W.T. Yin, S. Osher, D. Goldfarb, and F. Darbon. Bregman iterative algorithms for compressed sensing and related problems. *SIAM J. Img. Sci.*, 1:143–168, 2008.

[32] T. Goldstein and S. Osher. The split Bregman method for L1-regularized problems. *SIAM J. Img. Sci.*, 2:323 – 343, 2009.

[33] E. Esser. Applications of Lagrangian-based alternating direction methods and connections to split Bregman. Technical report, UCLA CAM Report 09-31, 2009.

[34] Y. Wang, J. Yang, W. Yin, and Y. Zhang. A new alternating menimization algorithm for total variation image reconstruction. *SIAM J. Img. Sci.*, 1:248–272, 2008.

[35] J. Yang, Y. Zhang, and W. Yin. An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. *SIAM J. Sci. Comput.*, 31(4):2842–2865, 2009.

[36] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-preserving variational multichannel image restoration. *SIAM J. Img. Sci.*, 2(2):569–592, 2009.

[37] S. Setzer. Splitting Bregman algorithm, Douglas-Rachford splitting and frame shrinkage. In *SSVM '09: Proceedings of the Second International Conference on Scale Space and Variational Methods in Computer Vision*, pages 464–476, Berlin, Heidelberg, 2009. Springer-Verlag.

[38] G. Steidl and T. Teuber. Removing multiplicative noise by Douglas-Rachford splitting methods. *J. Math. Imaging Vision Accepted*.

[39] X.-C. Tai and C. Wu. Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model. In *SSVM '09: Proceedings of the Second International Conference on Scale Space and Variational Methods in Computer Vision*, pages 502–513, Berlin, Heidelberg, 2009. Springer-Verlag.

[40] C. Wu and X.-C. Tai. Augmented Lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models. Technical report, UCLA CAM Report 09-76, 2009.

[41] C. Wu, J. Zhang, and X.-C. Tai. Augmented lagrangian method for total variation restoration with non-quadratic fidelity. Technical report, UCLA CAM Report 09-82, 2009.

[42] J. Koko and S. Jehan-Besson. An augmented Lagrangian method for $\mathrm{TV_g} + l^1$-norm minimization. Technical report, Research Reort LIMOS/RR-09-07, 2009.

[43] J.-F. Aujol. Some first-order algorithms for total variation based image restoration. *J. Math. Imaging Vis.*, 34:307–327, 2009.

[44] E. Esser, X Zhang, and T. Chan. A general framework for a class of first order primal-dual algorithms for TV minimization. Technical report, UCLA CAM Report 09-67, 2009.

[45] T. Rahman, X.-C. Tai, and S. Osher. A TV-Stokes denoising algorithm. In *Scale Space and Variational Methods in Computer Vision*, pages 473–482. Springer, Heidelberg, 2007.

[46] C. A. Elo, A. Malyshev, and T. Rahman. A dual formulation of the TV-Stokes algorithm for image denoising. In *SSVM '09: Proceedings of the Second International Conference on Scale Space and Variational Methods in Computer Vision*, pages 307–318, Berlin, Heidelberg, 2009. Springer-Verlag.

[47] S. Masnou and J.-M. Morel. Level lines based disocclusion. In *Proc. IEEE Int. Conf. on Image Processing, Chicago, IL*, pages 259–263, 1998.

[48] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. Image inpainting. In *Computer Graphics, SIGGRAPH 2000*, pages 417–424, 2000.

[49] T. F. Chan and J. Shen. Mathematical models for local nontexture inpaintings. *SIAM J. Appl. Math*, 62(3):1019–1043, 2002.

[50] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera. Filling-in by joint interpolation of vector fields and gray levels. *IEEE Trans. Image Process.*, 10(8):1200–1211, 2001.

[51] M. Bertalmio and A. L. Bertozzi and G. Sapiro. Navier-Stokes, fluid dynamica, and image and video inpainting. In *Proc. Conf. Comp. Vision Pattern Rec.*, pages 355–362, 2001.

[52] T. F. Chan, S.-H. Kang, and J. Shen. Euler's elastica and curvature based inpaintings. *SIAM J. Appl. Math.*, 63(2):564–594, 2002.

[53] T. F. Chan and J. Shen. Nontexture inpainting by curvature driven diffusion (CDD). *J. Visul Comm. Image Rep.*, 12:436–449, 2001.

[54] M. Bertalmio. Strong-continuation, contrast-invariant inpaing with a third-order optimal PDE. *IEEE Trans. Image Process.*, 15(7):1934–1938, 2006.

[55] P. Kornprobst and G. Aubert. Explicit reconstruction for image inpainting. In *Research Report N. 5905 INRIA*, 2006.

[56] M. Lysaker, S. Osher, and X.-C. Tai. Noise removal using smoothed normals and surface fitting. *IEEE Trans. Image Process.*, 13(10):1345–1357, 2004.

[57] F. Dong, Z. Liu, D. Kong, and K. Liu. An improved LOT model for image restoration. *J. Math. Imag. Vis.*, 34:89–97, 2009.

[58] X.-C. Tai, S. Osher, and R. Holm. Image inpainting using TV-Stokes equation. In *Image Processing Based on Partial Differential Equations*, pages 3–22. Springer, Heidelberg, 2006.

[59] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(4):439–451, 1988.

[60] N. Petrovic, I. Cohen, B. J. Frey, R. Koetter, and T. S. Huang. Enforcing integrability for surface reconstruction algorithms using belief propagation in graphical models. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 743–748, 2001.

[61] R. Basri, D. Jacobs, and I. Kemelmacher. Photometric stereo with general, unknown lighting. *Int. J. Comput. Vis.*, 72:239–257, 2007.

[62] D. Reddy, A. Agrawal, and R. Chellappa. Enforcing integrability by error correction using l^1-minimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2350–2357, 2009.

[63] W.G. Litvinov, T. Rahman, and X.-C. Tai. A modified TV-Stokes model for image processing. Technical report, Preprints - Herausgeber: Institut für Mathematik der Universität Augsburg 2009-25, 2009.

[64] B. K. P. Horn and M. J. Brooks. The variational approach to shape from shading. *Comput. Vis. Graph. Image Process.*, 33:174–208, 1986.

[65] Y. Ohtake, A. Belyaev, and I. Bogaevski. Mesh regularization and adaptive smoothing. *Comput.-Aided Design*, 33:789–800, 2001.

[66] S. F. Johnston. Lumo: illumination for cel animation. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 45–52, New York, NY, USA, 2002. ACM.

[67] T. Tasdizen, R. Whitaker, P. Burchard, and S. Osher. Geometric surface processing via normal maps. *ACM Trans. Graph. (TOG)*, 22:1012–1033, 2003.

[68] T.-P. Wu, C.-K. Tang, M.S. Brown, and H.-Y. Shum. Shapepalettes: Interactive normal transfer visa sketching. *ACM Transactions on Graphics volumn 26 Issue 307*, page Article No.44, 2007.

[69] T.-P. Wu, S. Sun, C.-K. Tang, and H.-Y. Shum. Interactive normal reconstruction from a sigle image. *ACM Transactions on Graphics volumn 27 Issue 508*, page Article No.119, 2008.

[70] H. S. Ng, T.-P. Wu, and C.-K. Tang. Surface-from-gradients with incomplete data for single view modeling. In *ICCV07*, pages 1–8, 2007.

[71] D. Terzopoulos. The computation of visible-surface representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(4):417–438, 1988.

[72] F. Catté, P. L. Lions, J.-M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29(1):182–193, 1992.

[73] J. Weickert. Coherence-enhancing diffusion filtering. *Int. J. Comput. Vis.*, 31:111–127, 1999.

[74] J. Hahn, B. Sofia, X.-C. Tai, and A. M. Bruckstein. On orientation-matching minimization TV-Stokes equation image denoising and image inpainting. *Accepted to Int. J. Comput. Vis.*, 2010.

[75] R. Glowinski and P. L. Tallec. *Augmented Lagrangians and Operator-Splitting Methods in Nonlinear Mechanics*. SIAM, Philadelphia, 1989.

[76] J. Weickert, B. M. ter Harr Romeny, and M. A. Viergever. Efficient and reliable schemes for nonlinear diffusion filtering. *IEEE Trans. Image Process.*, 7:398–410, 2001.

[77] Y. Meyer. *Oscillating patterns in image processing and nonlinear evolution equations.* AMS, Providence, RI, 2002.

[78] S. Osher, A. Solé, and L. Vese. Image decomposition and restoration using total variation minimization and the h^{-1} norm. *SIAM J. Multiscale Model. Simul.*, 1(3):349–370, 2002.

[79] J.-F. Aujol, G. Aubert, L. Blanc-Fáraud, and A. Chambolle. Image decomposition into a bounded variation component and an oscillating component. *J. Math. Imag. Vis.*, 22:71–88, 2005.

[80] P. Kovesi. Shapelets correlated with surface normals produce surfaces. In *ICCV05*, pages 994–1001, 2005.