Variational Structure-Texture Image Decomposition on Manifolds

Xiaoqun Wu[a], Jianmin Zheng[a], Chunlin Wu[b], Yiyu Cai[a]

[a] *Nanyang Technological University, Singapore*
[b] *National University of Singapore, Singapore*

Abstract

This paper considers the problem of decomposing an image defined on a manifold into a structural component and a textural component. We formulate such decomposition as a variational problem, in which the total variation energy is used for extracting the structural part and based on the properties of texture one of three norms, L^2, L^1 and G, is used in the fidelity term for the textural part. While L^2 and G norms are used for texture of no *a-prior* knowledge or oscillating pattern, L^1 norm is used for structural or sparse texture. We develop efficient numerical methods to solve the proposed variational problems using augmented Lagrangian methods (ALM) when the manifold is represented by a triangular mesh. The contributions of the paper are twofold: (1) We adapt the variational structure-texture image decomposition to manifolds, which takes the intrinsic property of manifolds into account. The non-quadratic fidelity terms with L^1 and G norms are extended to 3D triangular meshes for the first time. (2) We show how to efficiently tackle the variational problems with non-linearity/non-differentiability terms by iteratively solving some sub-problems that either have closed form solutions or are to solve linear equations. We demonstrate the effectiveness of the pro-

posed methods with examples and applications in detail enhancement and impulsive noise removal.

Keywords: Image decomposition, Manifolds, Structure-Texture, L^2-norm, L^1-norm, G-norm

---

1. Introduction

As one form of the task of decomposing signals into simpler signals of more coherent origin, the decomposition of an image into its structural and textural components is an important inverse problem in image processing. In general, the structural component is piecewise smooth, corresponding to the main large objects in the image, and the textural component captures the fine scale-details in the image, usually with some periodicity or oscillatory nature. When the textural component is noise, the decomposition process becomes image restoration or denoising. However, the decomposition is not just a low-pass/high-pass image denoising since both the structural and textural components may contain high and low frequencies [1, 2]. It has been shown that this decomposition is useful for HDR tone mapping [3, 4], flash/no-flash image fusion [5, 6] and image editing [7], and fundamental for image inpainting [8], image matching, compression, segmentation and classification [9].

1.1. Image decomposition

Many methods have been developed for decomposing images defined on a planar domain. Basically they can be classified into three categories. The first one is to use some filters such as linear/non-linear filters, bilateral filters and weighted least square filters. The linear filters are simple, but are known to produce halo artifacts near edges. The artifacts may be reduced

by using non-linear edge-preserving smoothing filters, examples of which are anisotropic diffusion [10], bilateral filters [11, 12] and weighed least square filters [4]. Filters are actually approximate solutions to some variational optimization problems.

The second category is to use total variation (TV), which is inspired by the TV regularization for image denoising and restoration [13]. The TV regularization has become a popular method for solving a wide variety of inverse problems in image processing due to the good edge-preserving property [14, 15, 16, 17, 18, 19]. In [15], several energy functionals in the form of TV plus a fidelity term for image decomposition are analyzed. In particular, L^2 and G norms [1, 2] are suggested for textures of no *a-prior* knowledge or oscillating pattern and L^1 norm is suggested for structural texture. Due to the non-differentiability and non-linearity of the total variation and the L^1 and G norms, sophisticated numerical methods are needed to efficiently solve the models. Examples include dual methods of CGM [20, 21], split Bregman iteration [22], as well as split-and-penalty based methods [23, 24], and augmented Lagrangian methods [25, 26].

The third category is to use sparse signal processing techniques. For example, the approach based on a combination of the Basis Pursuit Denoising (BPDN) and the TV regularization is proposed in [27]. Its basic idea is the use of two appropriate dictionaries, which are designed for sparse representations of structural and textural components, respectively. How to choose the proper dictionaries for a given image is a challenging problem.

1.2. Image processing on manifolds

Manifold surfaces are common in computer graphics and vision. Especially with advance of 3D data acquisition technology, the captured data often contain both 3D geometry and color information [28]. There has been increasing interest in studying image processing on surfaces or manifolds in the fields of computer vision, medical imaging, urban reconstruction and 3D cartoon. Some image processes such as image segmentation and restoration have been extended to manifolds [29, 25, 30]. However, the problem of decomposing data on manifolds has not received much attention. Moreover, many image processes on surfaces are considered on implicit representations, which define a surface as a zero level set of a signed distance function on an Euclidean domain or a narrow band of the given surface. The differential operators on surfaces are approximated by combining the standard Euclidean differential operators with the projection along the normal direction [31, 32]. In practice, it is not easy to obtain the implicit representation of a surface with complicated geometry or topology structure.

Triangular meshes are a popular representation in many applications. To facilitate data processing on triangular meshes, one approach is to parameterize the mesh first and then differential operators can be computed under the corresponding parametrization [33, 34, 35]. However, the computation of parametrization itself is a complicated task, especially for a surface with complicated structure or high genus. Recently, the image processing problems are formulated directly on meshes [36, 37, 29, 38, 30]. Fast convex optimization techniques for TV based methods are discussed in [38], where the split Bregman iteration is used for TV denoising of gray images and Chan-Vase

segmentation.

1.3. Our work

In this paper, we extend TV-based image decomposition models to manifolds represented by triangular meshes. Different fidelity terms, L^2, L^1 and G norms, are chosen according to the types of texture of the input image. L^2 norm is designed for extracting low-amplitude texture including Gaussian noise. L^1 norm preserves contrast and geometric structure and is thus suitable for structural or sparse textures such as impulsive noise. G norm is introduced for the texture of oscillatory pattern. We discretize the models directly on triangular meshes, which avoids parameterization but takes the intrinsic geometry of the mesh into account. The final formulations are non-linear and non-differentiable. We propose efficient numerical methods to solve these models.

The main contributions of the paper are:

- We adapt the multichannel variational image decomposition models to manifolds. Different fidelity terms with L^2, L^1 and G norms are considered in the variational models for different texture types. The challenging problems of L^1 and G norms for multichannel images are extended to triangular meshes for the first time.

- We develop three augmented Lagrangian methods based algorithms to solve the proposed variational decomposition models. The algorithms work by iterating several procedures of solving some sub-problems. Since the sub-problems either have closed form solutions or are to solve linear systems, our numerical methods are very simple and effective.

The paper is organized as follows. Section 2 presents three multichannel variational models with L^2, L^1 and G norms for image decomposition on manifolds and their discretization when the manifold is represented by a triangular mesh. Section 3 proposes numerical methods to solve the proposed three decomposition models. In Section 4, experiments are conducted to test the proposed methods. Two applications of the proposed decomposition are given in Section 5. Section 6 concludes the paper.

2. Image Decomposition Models on Manifolds

This section extends three variational models for image decomposition on a manifold and shows how to discretize the associated functional for a manifold represented by a triangular mesh.

2.1. Three variational models

Let M be a 2-dimensional Riemannian manifold and $\{U_\alpha, x_\alpha\}_{\alpha\in\mathcal{A}}$ be its differential structure where $\mathcal{A}$ is an index set. $\{U_\alpha, x_\alpha\}_{\alpha\in\mathcal{A}}$ can be viewed as a piecewise parameterization of M. We denote by (ξ^1, ξ^2) the local coordinates U_α. For any point $x \in x_\alpha(U_\alpha) \subset M$, the tangent space T_xM is spanned by $\{\frac{\partial x_\alpha}{\partial \xi^1}, \frac{\partial x_\alpha}{\partial \xi^2}\}$. The differential structure $\{U_\alpha, x_\alpha\}_{\alpha\in\mathcal{A}}$ gives a Riemannian metric tensor $\mathbb{G}$ on M:

$$\mathbb{G} = \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} = \begin{pmatrix} \frac{\partial x_\alpha}{\partial \xi^1}\cdot\frac{\partial x_\alpha}{\partial \xi^1} & \frac{\partial x_\alpha}{\partial \xi^1}\cdot\frac{\partial x_\alpha}{\partial \xi^2} \\ \frac{\partial x_\alpha}{\partial \xi^2}\cdot\frac{\partial x_\alpha}{\partial \xi^1} & \frac{\partial x_\alpha}{\partial \xi^2}\cdot\frac{\partial x_\alpha}{\partial \xi^2} \end{pmatrix} \tag{1}$$

where g_{11}, g_{12} (or g_{21}) and g_{22} are coefficients of the first fundamental form. The tensor $\mathbb{G}$ is symmetric and positive definite. It determines a bilinear

form on the tangent space T_xM. One can calculate the inner product of any two vectors on T_xM using the bilinear form.

An image defined on manifold M is a mapping $\mathbf{f} : M \to R^m$. When $m = 1$, it is a gray image; and when $m = 3$, it is a color image. The structure-texture image decomposition on a manifold is to decompose $\mathbf{f}$ into $\mathbf{u}$ and $\mathbf{v}$ such that

$$\mathbf{f} = \mathbf{u} + \mathbf{v} \tag{2}$$

where $\mathbf{u} = (u^1, u^2, \cdots, u^m)^T$ is a function defined on M, which is usually piecewise constant and conveys the geometric information; and $\mathbf{v} = (v^1, v^2, \cdots, v^m)^T$ is another function defined on M, encoding textural information. One way to implement such decomposition is to find $\mathbf{u}$ and $\mathbf{v}$ satisfying

$$\inf_{\{(\mathbf{u},\mathbf{v})|\mathbf{f}=\mathbf{u}+\mathbf{v}\}} \int_M |\nabla_M \mathbf{u}| dM + \lambda \|\mathbf{v}\|_{fid} \tag{3}$$

where the energy functional is a combination of a regularization term and a fidelity term with a tuning parameter $\lambda \in R$ balancing the two terms.

The regularization term $\int_M |\nabla_M u| dM$ is a total variation of $\mathbf{u}$ on M where dM is the manifold's element measure, ∇_M is the intrinsic gradient on M, defined by

$$\nabla_M \mathbf{u} = \left(\frac{\partial x_\alpha}{\partial \xi^1}, \frac{\partial x_\alpha}{\partial \xi^2} \right) \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial \mathbf{u}}{\partial \xi^1} \\ \frac{\partial \mathbf{u}}{\partial \xi^2} \end{pmatrix} \tag{4}$$

with $\mathbf{u}(x) = \mathbf{u}(x_\alpha(\xi^1, \xi^2))$, and then

$$|\nabla_M \mathbf{u}| = \sqrt{\left(\frac{\partial \mathbf{u}}{\partial \xi^1}, \frac{\partial \mathbf{u}}{\partial \xi^2} \right) \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix}^{-1} \begin{pmatrix} \frac{\partial \mathbf{u}}{\partial \xi^1} \\ \frac{\partial \mathbf{u}}{\partial \xi^2} \end{pmatrix}}. \tag{5}$$

The total variation regularization enables good edge preservation.

The fidelity term $\|\mathbf{v}\|_{fid}$ is formulated by L^2, L^1 and G norms on manifold M to fit three types of textures: low-amplitude texture, sparse structure texture (such as pepper and salt noise), and oscillating patterns, respectively. The details of the three norms are elaborated below.

- The L^2-norm variational image decomposition on manifold M is to find the solution of

$$\inf_{\{(\mathbf{u},\mathbf{v})|\mathbf{f}=\mathbf{u}+\mathbf{v}\}} \left(\int_M |\nabla_M \mathbf{u}| dM + \lambda \|\mathbf{v}\|_{2,M}^2 \right) \tag{6}$$

  where $\|\mathbf{v}\|_{2,M}^2 = \int_M \langle \mathbf{v}, \mathbf{v} \rangle dM$ with $\langle , \rangle$ denoting the inner product in Euclidean space, R^m. This is an extension of the famous ROF model [13] from a plane domain to a manifold. The ROF model was proposed initially for denoising, which preserves well the edges of the input image and removing most of the noise.

- The L^1-norm variational image decomposition on manifold M is to find the solution of

$$\inf_{\{(\mathbf{u},\mathbf{v})|\mathbf{f}=\mathbf{u}+\mathbf{v}\}} \left(\int_M |\nabla_M \mathbf{u}| dM + \lambda \|\mathbf{v}\|_{1,M} \right) \tag{7}$$

  where $\|\mathbf{v}\|_{1,M} = \int_M \sqrt{\langle \mathbf{v}, \mathbf{v} \rangle} dM$. Compared to the ROF model, the L^1-norm model does not erode geometric structure and is particularly well suited to remove salt and pepper noise. However, L^1-norm is highly non-linear and non-differentiable, which makes it difficult to solve the problem quickly and efficiently.

- The G-norm variational image decomposition on manifold M is to find

the solution of

$$\inf_{\{(\mathbf{u},\mathbf{v})|\mathbf{f}=\mathbf{u}+\mathbf{v}\}} \left(\int_M |\nabla_M \mathbf{u}| dM + \lambda \|\mathbf{v}\|_{G,M} \right) \tag{8}$$

where $\|\mathbf{v}\|_{G,M}$ is the G-norm on manifold M. The G-norm was initially suggested by Meyer for oscillations [1]. Here we adapt it on manifolds. Let $\mathbf{v}$ be written as

$$\begin{array}{rl} \mathbf{v} = div_M(\mathbf{g}) = & \frac{1}{\sqrt{det(\mathbb{G})}} \frac{\partial}{\partial \xi^1} \left(\sqrt{det(\mathbb{G})} g^1 \right) \\ + & \frac{1}{\sqrt{det(\mathbb{G})}} \frac{\partial}{\partial \xi^2} \left(\sqrt{det(\mathbb{G})} g^2 \right) \end{array} \tag{9}$$

for a vector $\mathbf{g} = g^1 \frac{\partial x_\alpha}{\partial \xi^1} + g^2 \frac{\partial x_\alpha}{\partial \xi^2}$. Then

$$\begin{array}{rl} \|\mathbf{v}\|_{G,M} = \inf_{\mathbf{g}} \Big\{ \sup_{x \in M} |\mathbf{g}| : & \mathbf{v} = div_M(\mathbf{g}), \\ & \mathbf{g} = g^1 \frac{\partial x_\alpha}{\partial \xi^1} + g^2 \frac{\partial x_\alpha}{\partial \xi^2} \Big\} \end{array} \tag{10}$$

with

$$|\mathbf{g}| = \sqrt{(g^1(x), g^2(x)) \begin{pmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{pmatrix} \begin{pmatrix} g^1(x) \\ g^2(x) \end{pmatrix}}.$$

Such $\mathbf{v}$ may have large oscillations with nevertheless a small norm. Thus the G norm is well-suited to capture the oscillations of a function in an energy minimization method.

However, a G-norm minimization problem is very difficult to solve. Note that there was no numerical scheme provided in [1]. Vese and Osher [2] devised a variant of the G-norm decomposition based on L^p norms and gave a numerical scheme using Euler-Lagrange equations. The variant of the G-norm decomposition on manifold M can then be

$$\begin{array}{l} \inf_{\mathbf{f}=\mathbf{u}+div_M(\mathbf{g})} \Big(\int_M |\nabla_M \mathbf{u}| dM + \kappa \left[\int_M |\mathbf{g}|^p dM \right]^{\frac{1}{p}} \\ \qquad + \lambda \|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|_{2,M}^2 \Big) \end{array} \tag{11}$$

In practice, it is often to choose $p = 1$. In this paper, we only consider the situation of $p = 1$, which is an approximation of the G-norm decomposition.

2.2. Discretization

Now we consider a manifold represented by a triangular mesh. Triangular meshes are a common shape representation in geometric modeling and computer graphics. Without ambiguity, we also use M to denote a triangular mesh surface of arbitrary topology in R^3 with no degenerate triangles. The set of vertices and the set of face triangles of M are denoted by $\{v_i : i = 0, 1, ..., N_v - 1\}$, $\{\tau_i : i = 0, 1, ..., N_\tau - 1\}$, respectively, where N_v, N_τ are the numbers of vertices and triangles.

As in the finite elements literature, for each vertex v_i, we define a piecewise linear basis functions $\phi_i : M \to R$ such that $\phi_i(v_j) = \delta_{ij}$, $i, j = 0, 1, ..., N_v - 1$, where δ_{ij} is the Kronecker delta. The vector valued field $\mathbf{u} = \{\mathbf{u}^1, \mathbf{u}^2, \cdots, \mathbf{u}^m\}$ with $\mathbf{u}^i = (u_0^i, u_1^i, \cdots, u_{N_v-1}^i)$ is defined on all vertices of the triangular mesh M. Suppose $\mathbf{u}$ reaches values $\mathbf{u}_j = (u_j^1, u_j^2, \cdots, u_j^m)$ at vertex $v_j, j = 0, 1, \cdots, N_v - 1$. We can use $\{\phi_i : i = 0, 1, \cdots, N_v - 1\}$ to construct a piecewise linear interpolating function on M, which naturally extends $\mathbf{u}$ on M. That is, $\mathbf{u}(x) = \sum_{i=0}^{N_v-1} \mathbf{u}_i \phi_i(x)$ for any $x \in M$.

We define two Hilbert spaces $\mathbf{V}_M^m = \underbrace{V_M \times V_M \times \cdots \times V_M}_{m}$ with $V_M = R^{N_v}$ and $\mathbf{Q}_M^m = \underbrace{Q_M \times Q_M \times \cdots \times Q_M}_{m}$ with $Q_M = R^{3 \times N_\tau}$. The elements in the linear space $\mathbf{V}_M^m$ are given by values at vertices of M. For example, $\mathbf{u} = (\mathbf{u}^1, \mathbf{u}^2, \cdots, \mathbf{u}^m) \in \mathbf{V}_M^m$ with $\mathbf{u}^i = (u_0^i, u_1^i, \cdots, u_{N_v-1}^i) \in V_M$ means that $\mathbf{u}$ takes value $\mathbf{u}_j = (u_j^1, u_j^2, \cdots, u_j^m)$ at vertex v_j for j. By the basis functions

$\{\phi_i : i = 0, 1, \cdots, N_v - 1\}$, $\mathbf{V}_M^m$ is isomorphic to the space of all piecewise linear vector-valued functions on M. We denote by Q_M the set of piecewise constant vector fields on M. The linear space $\mathbf{Q}_M^m$ is the product of m Q_M's. Thus an m-channel image $\mathbf{f} = (\mathbf{f}^1, \mathbf{f}^2, \cdots, \mathbf{f}^m)$ on M, where $\mathbf{f}^i = (f_0^i, f_1^i, \cdots, f_{N_v-1}^i)$, is an element of $\mathbf{V}_M^m$, and its intrinsic gradient $\nabla_M \mathbf{f} = (\nabla_M \mathbf{f}^1, \nabla_M \mathbf{f}^2, \cdots, \nabla_M \mathbf{f}^m)$ is an element of $\mathbf{Q}_M^m$.

We can equip the spaces $\mathbf{V}_M^m$ and $\mathbf{Q}_M^m$ with inner products and norms. For $\mathbf{u}, \mathbf{v} \in \mathbf{V}_M^m$, we define the inner product of $\mathbf{u}$ and $\mathbf{v}$ to be

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\mathbf{V}_M^m} = \sum_{j=0}^{N_v-1} s_j \sum_{i=1}^{m} u_j^i v_j^i \tag{12}$$

where $s_j = \sum_{\tau \in D_1(j)} \frac{1}{3} s_\tau$ with s_τ as the area of the triangle τ and $D_1(j)$ is the set of triangles having v_j as one of their vertices. The L^2 and L^1-norms of an element $\mathbf{u} \in \mathbf{V}_M^m$ are

$$\|\mathbf{u}\|_{2,\mathbf{V}_M^m}^2 = \langle \mathbf{u}, \mathbf{u} \rangle_{\mathbf{V}_M^m} \tag{13}$$

and

$$\|\mathbf{u}\|_{1,\mathbf{V}_M^m} = \sum_{j=0}^{N_v-1} s_j \sqrt{\sum_{i=1}^{m} (u_j^i)^2}, \tag{14}$$

respectively. Similarly, for $\mathbf{p}, \mathbf{q} \in \mathbf{Q}_M^m$, we define

$$\langle \mathbf{p}, \mathbf{q} \rangle_{\mathbf{Q}_M^m} = \sum_{\tau} s_\tau \sum_{i=1}^{m} \langle \mathbf{p}_\tau^i, \mathbf{q}_\tau^i \rangle \tag{15}$$

where $\langle \mathbf{p}_\tau^i, \mathbf{q}_\tau^i \rangle$ is the conventional inner product in R^3. The L^2 and L^1-norms of an element $\mathbf{p} \in \mathbf{Q}_M^m$ are

$$\|\mathbf{p}\|_{2,\mathbf{Q}_M^m}^2 = \langle \mathbf{p}, \mathbf{p} \rangle_{\mathbf{Q}_M^m} \tag{16}$$

and

$$\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} = \sum_{\tau} s_\tau \sqrt{\sum_{i=1}^{m} \langle \mathbf{p}_\tau^i, \mathbf{p}_\tau^i \rangle}, \tag{17}$$

respectively.

The gradient operator ∇_M on M can also be viewed as a mapping from V_M to Q_M or from $\mathbf{V}_M^m$ to $\mathbf{Q}_M^m$. For instance, given $\mathbf{u} \in \mathbf{V}_M^m$, corresponding to a piecewise linear function on M, its gradient is

$$\nabla_M \mathbf{u} = \sum_{i=0}^{N_v - 1} \mathbf{u}_i \nabla_M \phi_i \in \mathbf{Q}_M^m \tag{18}$$

and $\nabla_M \phi_i$ is a piecewise constant vector field on M. Referring to Figure 1, we introduce the local parametrization $\xi_{j,\tau} = (\xi_\tau^1, \xi_\tau^2) \in R^2$ for each triangle τ on manifold M with respect to vertex v_j such that for any point $v \in \tau$, we have $v = v_i + \xi_\tau^1 (v_k - v_i) + \xi_\tau^2 (v_j - O)$ where O is the project of v_j on edge $v_i v_k$. Thus the tensor $\mathbb{G}$ at point v is

$$\mathbb{G} = \begin{pmatrix} (v_k - v_i) \cdot (v_k - v_i) & 0 \\ 0 & (v_j - O) \cdot (v_j - O) \end{pmatrix}$$

and then

$$\nabla_M \phi_j = \frac{v_j - O}{\|v_j - O\|^2}. \tag{19}$$

The divergence operator $div_M : \mathbf{Q}_M^m \to \mathbf{V}_M^m$, for $\mathbf{p} \in \mathbf{Q}_M^m$, is given by

$$(div_M \mathbf{p})_{v_i} = -\frac{1}{s_i} \sum_{\tau, v_i \prec \tau} s_\tau \langle \mathbf{p}_\tau, (\nabla_M \phi)_\tau \rangle$$

where $v_i \prec \tau$ means that v_i is a vertex of triangle τ.

With all the above formulae, we can write the discrete versions of our L^2, L^1 and G-norm variational image decomposition models (6,7,11) as follows:

$$\min_{\mathbf{f}=\mathbf{u}+\mathbf{v}} \left(\|\nabla_M \mathbf{u}\|_{1,\mathbf{Q}_M^m} + \lambda \|\mathbf{v}\|_{2,\mathbf{V}_M^m}^2 \right), \tag{20}$$

$$\min_{\mathbf{f}=\mathbf{u}+\mathbf{v}} \left(\|\nabla_M \mathbf{u}\|_{1,\mathbf{Q}_M^m} + \lambda \|\mathbf{v}\|_{1,\mathbf{V}_M^m} \right), \tag{21}$$

$$\begin{aligned} \min_{\mathbf{f}=\mathbf{u}+div_M(\mathbf{g})} \big(& \|\nabla_M \mathbf{u}\|_{1,\mathbf{Q}_M^m} + \kappa \|\mathbf{g}\|_{1,\mathbf{Q}_M^m} \\ & + \lambda \|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|_{2,\mathbf{V}_M^m} \big) . \end{aligned} \tag{22}$$

3. Numerical Methods

The challenge of our variational optimization problems comes from the non-differentiability and non-linearity of the total variation term and the L^1 and G fidelity terms. These models are often solved by gradient descent methods such as isotropic and anisotropic diffusion approaches, which are usually quite slow. Designing fast solvers is an active research topic in numerical computation. Recently, various convex optimization techniques have been proposed to efficiently solve TV based optimization problems. In this section we adapt augmented Lagrangian methods [26, 25] to solve variational image decomposition on a triangulated surface. Augmented Lagrangian methods are a class of algorithms for solving constrained optimization problems. They are known as a good alternative to penalty methods in that they replace a constrained optimization problem by a series of unconstrained problems which have an additional term similar to a Lagrange multiplier in the unconstrained objective. The unconstrained objective in the ALM is the Lagrangian of the contrained problem with an additional penalty term called the augmentation.

3.1. TV-L^2

We introduce a new variable $\mathbf{p} \in \mathbf{Q}_M^m$ and reformulate the TV-L^2 problem of (20) to the following constrained minimization problem

$$\begin{aligned} &\min_{\mathbf{u}\in\mathbf{V}_M^m,\mathbf{p}\in\mathbf{Q}_M^m} \left\{ \|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \lambda\|\mathbf{f}-\mathbf{u}\|_{2,\mathbf{V}_M^m}^2 \right\} \\ &s.t. \quad \mathbf{p} = \nabla_M \mathbf{u} \end{aligned} \tag{23}$$

To solve this problem, we define the augmented Lagrangian functional

$$\begin{aligned} L(\mathbf{u},\mathbf{p};\mu) \quad &= \|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \lambda\|\mathbf{f}-\mathbf{u}\|_{2,\mathbf{V}_M^m}^2 \\ &+\langle\mu, \mathbf{p}-\nabla_M\mathbf{u}\rangle_{\mathbf{Q}_M^m} + \tfrac{r}{2}\|\mathbf{p}-\nabla_M\mathbf{u}\|_{2,\mathbf{Q}_M^m}^2 \end{aligned} \tag{24}$$

where $\mu \in \mathbf{Q}_M^m$ is the Lagrangian multiplier, r is a positive constant and $\frac{r}{2}\|\mathbf{p}-\nabla_M\mathbf{u}\|_{2,M}^2$ is the augmented term, and then consider the following saddle-point problem: Find $(\mathbf{u}^*,\mathbf{p}^*;\mu^*) \in \mathbf{V}_M^m \times \mathbf{Q}_M^m \times \mathbf{Q}_M^m$ such that

$$L(\mathbf{u}^*,\mathbf{p}^*;\mu) \le L(\mathbf{u}^*,\mathbf{p}^*;\mu^*) \le L(\mathbf{u},\mathbf{p};\mu^*)$$

for all $(\mathbf{u},\mathbf{p};\mu) \in \mathbf{V}_M^m \times \mathbf{Q}_M^m \times \mathbf{Q}_M^m$.

Similar to [30] and references therein, it can be shown that this saddle-point problem has at least one solution and all the saddle-points $(\mathbf{u}^*,\mathbf{p}^*;\mu^*)$'s have the same $\mathbf{u}^*$, which is the unique solution to the original problem (20). Based on the augmented Lagrangian method, we present an iterative algorithm to solve the saddle-point problem, which is described in Algorithm 1.

In Algorithm 1, we compute an approximate solution to the minimization problem (25) by iteratively solving two sub-problems: the **u**-sub problem and the **p**-sub problem. The number of inner iterations increases the accuracy of (25) and ensures the convergence of overall algorithm. When in image case, it is fully accuracy ($K \to \infty$) and rough accuracy ($K = 1$). Some convergence

Algorithm 1 ALM algorithm for TV-L^2 decomposition

Initialization: $\mathbf{u}^{(0)} = \mathbf{p}^{(0)} = \mu^{(0)} = \epsilon = 0$.

while $\epsilon < \epsilon_0$ **do**

Compute $(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)})$ as an (approximate) minimizer of the augmented Lagrangian functional with the Lagrangian multiplier $\mu^{(k)}$:

$$(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}) \approx \arg\min_{\mathbf{u},\mathbf{p}} L(\mathbf{u}, \mathbf{p}, \mu^{(k)}) \tag{25}$$

which is done by the following iterations:

Let $\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)}$.

for $l = 1, 2, \cdots, K$ **do**

compute $\mathbf{u}^{(k+1)} = \arg\min_{\mathbf{u}} L(\mathbf{u}, \mathbf{p}^{(k+1)}; \mu^{(k)})$;

compute $\mathbf{p}^{(k+1)} = \arg\min_{\mathbf{p}} L(\mathbf{u}^{(k+1)}, \mathbf{p}; \mu^{(k)})$;

end for

and then update

$$\mu^{(k+1)} = \mu^{(k)} + r(\mathbf{p}^{(k+1)} - \nabla_M \mathbf{u}^{(k+1)})$$

$\epsilon = \|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\|_{2,V_M^m}$, $k++$.

end while

analysis is given in [39]. As observed in [30] and also our experiments, a small value suffices. In this paper, we choose $K = 1$. In addition, the two sub-problems can be easily solved.

- The **u**-sub problem: Given **p**, solve

$$\begin{aligned}\min_{\mathbf{u}}\{\lambda\|\mathbf{f}-\mathbf{u}\|^2_{2,V_M^m} \quad &+\langle\mu^{(k)},\mathbf{p}-\nabla_M\mathbf{u}\rangle_{\mathbf{Q}_M^m}\\ &+\tfrac{r}{2}\|\mathbf{p}-\nabla_M\mathbf{u}\|^2_{2,\mathbf{Q}_M^m}\}\end{aligned} \tag{26}$$

  This is a quadratic programming problem, whose optimality condition is

$$2\lambda(\mathbf{f}-\mathbf{u}) + div_M(\mu^{(k)}) + r div_M(\mathbf{p}-\nabla_M\mathbf{u}) = 0,$$

  which turns to be a sparse linear problem and can be solved by various well-developed numerical packages.

- The **p**-sub problem: Given **u**, solve

$$\min_{\mathbf{p}}\{\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \langle\mu^{(k)},\mathbf{p}\rangle_{\mathbf{Q}_M^m} + \frac{r}{2}\|\mathbf{p}-\nabla_M\mathbf{u}\|^2_{2,\mathbf{Q}_M^m}\}$$

  This problem is decomposable and thus can be solved triangle-by-triangle. That is, for each triangle τ, we solve

$$\begin{aligned}\min_{\mathbf{p}_\tau}\{&\sqrt{\sum_{i=1}^m\langle\mathbf{p}_\tau^i,\mathbf{p}_\tau^i\rangle} + \sum_{i=1}^m\langle{\mu_\tau^i}^{(k)},\mathbf{p}_\tau^i\rangle_{\mathbf{Q}_M^m}\\ &+\frac{r}{2}\sum_{i=1}^m\langle\mathbf{p}_\tau^i-(\nabla_M\mathbf{u}^i)_\tau,\mathbf{p}_\tau^i-(\nabla_M\mathbf{u}^i)_\tau\rangle\}\end{aligned}$$

  By a simple geometric analysis, it can be found that the above problem has a closed form solution

$$\mathbf{p}_\tau = \max(0, 1-\frac{1}{r|\mathbf{w}_\tau|})\mathbf{w}_\tau \tag{27}$$

where $\mathbf{w}_\tau = (\nabla_M \mathbf{u})_\tau - \dfrac{\mu_\tau^{(k)}}{r}$.

3.2. TV-L^1

When the fidelity term is in L^1, we introduce two new variables $\mathbf{p} \in \mathbf{Q}_M^m$, $\mathbf{z} \in \mathbf{V}_M^m$, and reformulate the TV-L^1 problem to the following constrained optimization problem:

$$\begin{aligned} &\min_{\mathbf{u},\mathbf{p},\mathbf{z}} \{\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \lambda \|\mathbf{z}\|_{1,\mathbf{V}_M^m}\} \\ &s.t. \quad \mathbf{p} = \nabla_M \mathbf{u},\ \mathbf{z} = \mathbf{u} - \mathbf{f} \end{aligned} \tag{28}$$

We then define the following augmented Lagrangian functional

$$\begin{aligned} L(\mathbf{u}, \mathbf{p}, \mathbf{z}; \mu_p, \mu_z) &= \|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \lambda \|\mathbf{z}\|_{1,\mathbf{V}_M^m} \\ &+ \langle \mu_p, \mathbf{p} - \nabla_M \mathbf{u} \rangle_{\mathbf{Q}_M^m} + \langle \mu_z, \mathbf{z} - (\mathbf{u} - \mathbf{f}) \rangle_{\mathbf{V}_M^m} \\ &+ \tfrac{r_p}{2} \|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2 + \tfrac{r_z}{2} \|\mathbf{z} - (\mathbf{u} - \mathbf{f})\|_{2,\mathbf{V}_M^m}^2 \end{aligned}$$

where $\mu_p \in \mathbf{Q}_M^m$, $\mu_z \in \mathbf{V}_M^m$ are the Lagrangian multipliers, and r_p and r_z are positive numbers, and $\frac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2$ and $\frac{r_z}{2}\|\mathbf{z} - (\mathbf{u} - \mathbf{f})\|_{2,\mathbf{V}_M^m}^2$ are the augmented terms.

As in Section 3.1, we solve (28) by adapting the augmented Lagrangian method to find a solution to the saddle-point problem:

$$\min_{\mathbf{u},\mathbf{p},\mathbf{z}} \max_{\mu_p,\mu_z} L(\mathbf{u}, \mathbf{p}, \mathbf{z}; \mu_p, \mu_z)$$

which iteratively solves a series of unconstrained minimization problems. The details are given in Algorithm 2.

It can be seen that Algorithm 2 involves three sub problems: the $\mathbf{u}$-sub problem, the $\mathbf{p}$-sub problem, and the $\mathbf{z}$-sub problem. These sub-problems actually can be solved easily.

Algorithm 2 ALM algorithm for TV-L^1 decomposition

Initialization: $\mathbf{u}^{(0)} = \mathbf{z}^{(0)} = \mu_z^{(0)} = \mathbf{p}^{(0)} = \mu_p^{(0)} = \epsilon = 0$.

while $\epsilon < \epsilon_0$ **do**

Compute $(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{z}^{(k+1)})$ as an (approximate) minimizer of the augmented Lagrangian functional with the Lagrangian multipliers $\mu_p^{(k)}$ and $\mu_z^{(k)}$:

$$(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{z}^{(k+1)}) \approx \arg\min_{\mathbf{u},\mathbf{p},\mathbf{z}} L(\mathbf{u}, \mathbf{p}, \mathbf{z}; \mu_p^{(k)}, \mu_z^{(k)})$$

which is done by the following iterations:

Let $\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)}, \mathbf{z}^{(k+1)} = \mathbf{z}^{(k)}$.

for $l = 1, 2, \cdots, K$ **do**

compute

$\mathbf{u}^{(k+1)} = \arg\min_{\mathbf{u}} L(\mathbf{u}, \mathbf{p}^{(k+1)}, \mathbf{z}^{(k+1)}; \mu_p^{(k)}, \mu_z^{(k)})$;

$\mathbf{p}^{(k+1)} = \arg\min_{\mathbf{p}} L(\mathbf{u}^{(k+1)}, \mathbf{p}, \mathbf{z}^{(k+1)}; \mu_p^{(k)}, \mu_z^{(k)})$;

$\mathbf{z}^{(k+1)} = \arg\min_{\mathbf{p}} L(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{z}; \mu_p^{(k)}, \mu_z^{(k)})$;

end for

and then update

$$\begin{aligned} \mu_p^{(k+1)} &= \mu_p^{(k)} + r_p(\mathbf{p}^{(k+1)} - \nabla_M \mathbf{u}^{(k+1)}) \\ \mu_z^{(k+1)} &= \mu_z^{(k)} + r_z(\mathbf{z}^{(k+1)} - (\mathbf{u}^{(k+1)} - \mathbf{f})) \end{aligned}$$

$\epsilon = \|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\|_{2,V_M^m}$, $k++$.

end while

- The **u**-sub problem: Given **p** and **z**, solve

$$\min_{\mathbf{u}}\{\langle \mu_p^{(k)}, \mathbf{p} - \nabla_M \mathbf{u}\rangle_{\mathbf{Q}_M^m} + \langle \mu_z^{(k)}, \mathbf{z} - (\mathbf{u} - \mathbf{f})\rangle_{\mathbf{V}_M^m} + \frac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2 + \frac{r_z}{2}\|\mathbf{z} - (\mathbf{u} - \mathbf{f})\|_{2,\mathbf{V}_M^m}^2\}$$

whose optimality condition is

$$\begin{aligned} div_M(\mu_p^{(k)}) - \mu_z^{(k)} &+ r_p div_M(\mathbf{p} - \nabla_M \mathbf{u}) \\ &+ r_z(\mathbf{u} - (\mathbf{z} + \mathbf{f})) = 0. \end{aligned}$$

This is a system of sparse linear equations.

- The **p**-sub problem: Given **u** and **z**, solve

$$\min_{\mathbf{p}}\{\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \langle \mu_p^{(k)}, \mathbf{p} - \nabla_M \mathbf{u}\rangle_{\mathbf{Q}_M^m} + \frac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2\} \tag{29}$$

Similar to the **p**-sub problem in the TV-L^2 model, this problem is decomposable and has a closed form solution for each triangle τ:

$$\mathbf{p}_\tau = \max(0, 1 - \frac{1}{r_p|\mathbf{w}_\tau|})\mathbf{w}_\tau \tag{30}$$

where $\mathbf{w}_\tau = (\nabla_M \mathbf{u})_\tau - \frac{(\mu_p)_\tau^{(k)}}{r_p}$.

- The **z**-sub problem: Given **u** and **p**, solve

$$\min_{\mathbf{z}}\{\lambda\|\mathbf{z}\|_{1,\mathbf{V}_M^m} + \langle \mu_z^{(k)}, \mathbf{z} - (\mathbf{u} - \mathbf{f})\rangle_{\mathbf{V}_M^m} + \frac{r_z}{2}\|\mathbf{z} - (\mathbf{u} - \mathbf{f})\|_{2,\mathbf{V}_M^m}^2\}. \tag{31}$$

Fortunately, this problem has a closed form solution:

$$\mathbf{z} = \max(0, 1 - \frac{\lambda}{r_z|\mathbf{t}|})\mathbf{t}$$

where $\mathbf{t} = \mathbf{u} - \mathbf{f} - \frac{\mu_z^{(k)}}{r_z}$.

3.3. TV-G

To solve the minimization problem (22), we introduce two variables $\mathbf{p} \in \mathbf{Q}_M^m$, $\mathbf{h} \in \mathbf{Q}_M^m$ and convert the problem to

$$\begin{aligned} &\min_{\mathbf{u},\mathbf{g},\mathbf{h}} \{\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \kappa|\mathbf{h}|_{1,\mathbf{Q}_M^m} + \lambda\|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|_{2,\mathbf{V}_M^m}^2\} \\ &\quad s.t. \qquad \mathbf{p} = \nabla_M \mathbf{u}, \qquad \mathbf{h} = \mathbf{g} \end{aligned}$$

Similar to the approaches for TV-L^2 and TV-G, we solve this problem by searching for a solution of the saddle-point problem:

$$\min_{\mathbf{u},\mathbf{p},\mathbf{g},\mathbf{h}} \max_{\mu_p,\mu_h} L(\mathbf{u}, \mathbf{p}, \mathbf{g}; \mu_p, \mu_h)$$

with the augmented Lagrangian functional $L(\mathbf{u}, \mathbf{p}, \mathbf{g}; \mu_p, \mu_h)$ given by

$$\begin{aligned} &L(\mathbf{u}, \mathbf{p}, \mathbf{g}; \mu_p, \mu_h) = \|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \kappa\|\mathbf{h}\|_{1,\mathbf{Q}_M^m} \\ &+\lambda\|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|_{2,\mathbf{V}_M^m}^2 + \langle \mu_p, \mathbf{p} - \nabla_M \mathbf{u}\rangle_{\mathbf{Q}_M^m} + \\ &\langle \mu_h, \mathbf{h} - \mathbf{g}\rangle_{\mathbf{Q}_M^m} + \tfrac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2 + \tfrac{r_h}{2}\|\mathbf{h} - \mathbf{g}\|_{2,\mathbf{Q}_M^m}^2 \end{aligned}$$

where $\mu_p \in \mathbf{Q}_M^m$, $\mu_h \in \mathbf{Q}_M^m$ are the Lagrangian multipliers, r_p and r_h are positive constants, and $\frac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2$ and $\frac{r_h}{2}\|\mathbf{h} - \mathbf{g}\|_{2,\mathbf{Q}_M^m}^2$ are the augmented terms. The main steps are given in Algorithm 3.

Algorithm 3 contains four sub-problems: the **u**-sub problem, the **p**-sub problem, the **g**-sub problem, and the **h**-sub problem. They can be solved in the following ways.

- The **u**-sub problem: Given $\mathbf{p}, \mathbf{g}$ and $\mathbf{h}$, solve

$$\begin{aligned} &\min_{\mathbf{u}} \{\lambda\|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|_{2,\mathbf{V}_M^m}^2 \\ &+\langle \mu_p^{(k)}, \mathbf{p} - \nabla_M \mathbf{u}\rangle_{\mathbf{Q}_M^m} + \tfrac{r_p}{2}\|\mathbf{p} - \nabla_M \mathbf{u}\|_{2,\mathbf{Q}_M^m}^2\} \end{aligned}$$

Algorithm 3 ALM algorithm for TV-G decomposition

Initialization: $\mathbf{u}^{(0)} = \mathbf{p}^{(0)} = \mathbf{h}^{(0)} = \mathbf{g}^{(0)} = \mu_p^{(0)} = \mu_h^{(0)} = \epsilon = 0$.

while $\epsilon < \epsilon_0$ **do**

Compute $(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{g}^{(k+1)}, \mathbf{h}^{(k+1)})$ as an (approximate) minimizer of the augmented Lagrangian functional with the Lagrangian multipliers $\mu_p^{(k)}$ and $\mu_h^{(k)}$:

$$(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{g}^{(k+1)}) \approx \arg\min_{\mathbf{u},\mathbf{p},\mathbf{h},\mathbf{g}} L(\mathbf{u}, \mathbf{p}, \mathbf{h}, \mathbf{g}; \mu_p^{(k)}, \mu_h^{(k)})$$

which is done by the following iterations:

Let $\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)}, \mathbf{g}^{(k+1)} = \mathbf{g}^{(k)}, \mathbf{h}^{(k+1)} = \mathbf{h}^{(k)}$.

for $l = 1, 2, \cdots, K$ **do**

compute

$\mathbf{u}^{(k+1)} = \arg\min_{\mathbf{u}} L(\mathbf{u}, \mathbf{p}^{(k+1)}, \mathbf{g}^{(k+1)}, \mathbf{h}^{(k+1)}; \mu_p^{(k)}, \mu_h^{(k)})$;

$\mathbf{p}^{(k+1)} = \arg\min_{\mathbf{p}} L(\mathbf{u}^{(k+1)}, \mathbf{p}, \mathbf{g}^{(k+1)}, \mathbf{h}^{(k+1)}; \mu_p^{(k)}, \mu_h^{(k)})$;

$\mathbf{g}^{(k+1)} = \arg\min_{\mathbf{g}} L(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{g}, \mathbf{h}^{(k+1)}; \mu_p^{(k)}, \mu_h^{(k)})$;

$\mathbf{h}^{(k+1)} = \arg\min_{\mathbf{h}} L(\mathbf{u}^{(k+1)}, \mathbf{p}^{(k+1)}, \mathbf{g}^{(k+1)}, \mathbf{h}; \mu_p^{(k)}, \mu_h^{(k)})$;

end for

and then update

$$\begin{aligned} \mu_p^{(k+1)} &= \mu_p^{(k)} + r_p(\mathbf{p}^{(k+1)} - \nabla_M \mathbf{u}^{(k+1)}) \\ \mu_h^{(k+1)} &= \mu_h^{(k)} + r_h(\mathbf{h}^{(k+1)} - \mathbf{g}^{(k+1)}) \end{aligned}$$

$\epsilon = \|\mathbf{u}^{(k+1)} - \mathbf{u}^{(k)}\|_{2,V_M^m}$, $k++$.

end while

whose solution is

$$(2\lambda - r_p\triangle)\mathbf{u} = 2\lambda\mathbf{f} - div_M(2\lambda\mathbf{g} + \mu_p^{(k)} + r_p\mathbf{p})$$

where $\triangle = div_M \circ \nabla_M$.

- The $\mathbf{p}$-sub problem: Given $\mathbf{u}, \mathbf{g}$ and $\mathbf{h}$, solve

$$\min_{\mathbf{p}}\{\|\mathbf{p}\|_{1,\mathbf{Q}_M^m} + \langle\mu_p^{(k)}, \mathbf{p} - \nabla_M\mathbf{u}\rangle_{\mathbf{Q}_M^m} + \tfrac{r_p}{2}\|\mathbf{p} - \nabla_M\mathbf{u}\|^2_{2,\mathbf{Q}_M^m}\}$$

which has a closed form solution for each triangle τ:

$$\mathbf{p}_\tau = \max(0, 1 - \frac{1}{r_p|\mathbf{w}_\tau|})\mathbf{w}_\tau$$

where $\mathbf{w}_\tau = (\nabla_M\mathbf{u})_\tau - \dfrac{(\mu_p)_\tau^{(k)}}{r_p}$.

- The $\mathbf{g}$-sub problem: Given $\mathbf{p}, \mathbf{u}$ and $\mathbf{g}$, solve

$$\min_{\mathbf{u}}\{\lambda\|\mathbf{f} - \mathbf{u} - div_M(\mathbf{g})\|^2_{2,\mathbf{V}_M^m} + \langle\mu_h^{(k)}, \mathbf{h} - \mathbf{g}\rangle_{\mathbf{Q}_M^m} + \tfrac{r_h}{2}\|\mathbf{h} - \mathbf{g}\|^2_{2,\mathbf{Q}_M^m}\}$$

whose solution is

$$(r_h - 2\lambda(\nabla \circ div)_M)\mathbf{g} = r_h\mathbf{h} - 2\lambda(\nabla_M(\mathbf{f} - \mathbf{u})) + \mu_h^{(k)}.$$

- The $\mathbf{h}$-sub problem: Given $\mathbf{u}, \mathbf{p}$ and $\mathbf{g}$, solve

$$\min_{\mathbf{u}}\{\kappa\|\mathbf{h}\|_{1,\mathbf{Q}_M^m} + \langle\mu_h^{(k)}, \mathbf{h} - \mathbf{g}\rangle_{\mathbf{Q}_M^m} + \frac{r_h}{2}\|\mathbf{h} - \mathbf{g}\|^2_{2,\mathbf{Q}_M^m}\}$$

which has a closed form solution for each triangle τ:

$$\mathbf{h} = \max\{0, 1 - \frac{\kappa}{|\mathbf{l}|r_h}\}\mathbf{l}$$

where $\mathbf{l} = \mathbf{g} - \dfrac{\mu_h^{(k)}}{r_h}$.

4. Experiments

In this section, we conduct experiments to test our variational structure-texture image decomposition on manifolds. The proposed algorithms were implemented in C++ with CGAL data structure. The experiments are carried out under Windows 7 running on a PC with Inter Xeon Core 2.27GHz CPU and 4GB RAM. Table 1 reports the numbers of vertices and triangles of the mesh models used in the experiments. The selection of the parameters and the running time for these examples are listed in Table 2. The algorithms take only a few seconds to complete the decomposition.

First, we compare our method with conventional image decomposition for planar images. Our proposed method decomposes images on manifolds. Planes are a special case of manifolds. For an image defined on a plane domain, it can be decomposed by our algorithms if we represent the plane domain as a triangular mesh. Fig. 2 is an example that shows the L^2 and L^1 decomposition results generated by the conventional planar image decomposition algorithms [18, 26] and our proposed method. The running time is reported in Table 2. It can be found from the experimental results that both decomposition methods perform similarly for planar images.

Second, we apply our decomposition algorithm to a textured manifold surface. The manifold surface is a 3D triangular mesh (see Fig. 3). Our decomposition algorithm separates the texture from the structure image. It can be seen that with L^2 norm, the structural component is smoother overall while the main structures are still preserved due to the TV term, and with L^1 norm, finer structures can also be captured in the structural component.

Third, we demonstrate the effectiveness of G norm for oscillatory pattern

texture and L^1 for sparse signal details. Fig. 4 shows an image with oscillatory pattern. We apply TV-L^2, TV-L^1 and TV-G decomposition models to the input image. Here, for L^2 norm, $\lambda = 250, r_p = 0.04$; for L^1 norm, $\lambda = 250, r_p = 0.04, r_z = 200$; for G norm, $\lambda = 250, r_p = 0.04, r_h = 200, \kappa = 0.01$. The results show that the fidelity with G norm extracts the oscillation best. Fig. 5 is an example where the input image contains sparse signal details. The decomposition results show that L^1 norm is more adapted for sparse details. Here, for L^2 norm, $\lambda = 200, r_p = 0.015$; for L^1 norm, $\lambda = 100, r_p = 0.015, r_z = 150$; and for G norm, $\lambda = 200, r_p = 0.015, r_h = 300, \kappa = 0.01$.

5. Applications

5.1. Detail exaggeration

One application of the structure-texture image decomposition is in image editing on manifolds. Once the structural and textural components of an image are separated, one can edit both components separately. For example, one can simply scale the the textural component to generate exaggerated details. Fig. 6(a) is a textured cup. Its structural components extracted by the TV-L^2 and TV-L^1 decomposition methods are shown in Fig. 6(b) and (d). The structural components are composed with their corresponding scaled textural components, yielding the effects of detail exaggeration, which are shown in Fig. 6(c) and (e).

5.2. L^1 image denoising on manifolds

Another application of our proposed method is image denoising on manifolds. As mentioned earlier, image decomposition can be used as a method

for image denoising or restoration if the texture is noise. Paper [30] considers the TV-L^2 model for image restoration on manifolds with Gaussian noise. However, if the noise is impulsive, the TV-L^2 model may not work well. Our proposed TV-L^1 model provides a better solution, which is confirmed by the experiments. Figs. 7 shows the comparison of image denoising by our TV-L^1 method and the TV-L^2 model of [30]. The parameters and experimental statistics of these examples are given in Table 3. The SNRs (signal-to-noise ratios) used in the examples were computed by

$$SNR = 10 \log_{10} \frac{\|\mathbf{u} - E(\mathbf{u})\|_{V_M}^2}{\|\mathbf{u} - \mathbf{g}\|_{V_M}^2}$$

where $\mathbf{u}$ is the noise-free (clean) image, $\mathbf{g}$ is the observed or recovered image, and $E(\mathbf{u})$ is the average intensity of the image $\mathbf{u}$ on M. Both the visual effects and the quantitative results show that our TV-L^1 method outperforms the method of [30] for removing impulsive noise.

6. Conclusions

We have presented TV-based approaches for structure-texture image decomposition on manifolds. The image on a manifold is separated into a piecewise smooth structure component and a fine scale-detailed texture component. Unlike planar image decomposition, the intrinsic geometry of the manifold is taken into account in our proposed decomposition models. The TV regularization effectively preserves edges of the structure image. L^2, L^1 and G fidelity norms well adapt to different types of textures. We have also developed efficient numerical algorithms, based on augmented Lagrangian methods, to implement such variational image decomposition on a manifold

represented as a triangular mesh. The efficiency of the proposed methods are demonstrated by several examples and two applications.

References

[1] Y. Meyer, Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures, American Mathematical Society, Boston, MA, USA, 2001.

[2] L. A. Vese, S. J. Osher, Modeling textures with total variation minimization and oscillating patterns in image processing, Journal of Scientific Computing 19 (2003) 553–572.

[3] J. Tumblin, G. Turk, LCIS: a boundary hierarchy for detail-preserving contrast reduction, in: Annual Conference on Computer Graphics, 1999, pp. 83–90.

[4] Z. Farbman, R. Fattal, D. Lischinski, R. Szeliski, Edge-preserving decompositions for multi-scale tone and detail manipulation, ACM Trans. Graph. 27 (3) (2008) 67:1–67:10.

[5] G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, K. Toyama, Digital photography with flash and no-flash image pairs, ACM Trans. Graph. 23 (2004) 664–672.

[6] E. Eisemann, F. Durand, Flash photography enhancement via intrinsic relighting, ACM Trans. Graph. 23 (2004) 673–678.

[7] B. M. Oh, M. Chen, J. Dorsey, F. Durand, Image-based modeling and photo editing, in: Proceedings of the 28th annual conference on Com-

puter graphics and interactive techniques, SIGGRAPH '01, ACM, New York, NY, USA, 2001, pp. 433–442.

[8] M. Bertalmio, L. Vese, G. Sapiro, S. Osher, Simultaneous structure and texture image inpainting, IEEE Transactions on Image Processing 12 (2003) 882 – 889.

[9] J.-F. Aujol, T. F. Chan, Combining geometrical and textured information to perform image classification, J. Visual Communication and Image Representation 17 (5) (2006) 1004–1023.

[10] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell. 12 (1990) 629–639.

[11] C. Tomasi, R. Manduchi, Bilateral filtering for gray and color images, in: Proceedings of the Sixth International Conference on Computer Vision, 1998.

[12] Q. Liu, J. Luo, Y. Zhu, Adaptive image decomposition by improved bilateral filter, International Journal of Computer Applications 23 (7) (2011) 16–22.

[13] L. I. Rudin, S. Osher, E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D 60 (1992) 259–268.

[14] E. J. Candès, F. Guo, New multiscale transforms, minimum total variation synthesis: applications to edge-preserving image reconstruction, Signal Processing 82 (2002) 1519–1543.

[15] J.-F. Aujol, G. Gilboa, T. Chan, S. Osher, Structure-texture image decomposition–modeling, algorithms, and parameter selection, International Journal of Computer Vision 67 (2006) 111–136.

[16] J. Oliveira, J. Nioucas-Dias, M. Figueiredo, Adaptive total variation image deblurring: A majorization-minimization approach, Signal Processing 89 (2009) 1683–1693.

[17] Y. Lin, N. Wohlberg, H. Guo, UPRE method for total variation parameter selection, Signal Processing 90 (2010) 2546–2551.

[18] J. Zhang, C. Wu, Fast optimization for multichannel total variation minimization with non-quadratic fidelity, Signal Processing 91 (2011) 1933–1940.

[19] A. Chambolle, P.-L. Lions, Image recovery via total variation minimization and related problems, Numerische Mathematik 76 (2) (1997) 167–188.

[20] T. F. Chan, G. H. Golub, P. Mulet, A nonlinear primal-dual method for total variation-based image restoration, SIAM J. Sci. Comput. 20 (6) (1999) 1964–1977.

[21] A. Chambolle, An algorithm for total variation minimization and applications, Journal of Mathematical Imaging and Vision 20 (2004) 89–97.

[22] T. Goldstein, S. Osher, The Split Bregman Method for L^1-regularized problems, SIAM J. Img. Sci. 2 (2009) 323–343.

[23] Y. Wang, W. Yin, Y. Zhang, A fast algorithm for image deblurring with total variation regularization, Tech. rep., Department of Computational and Applied Mathematics, Rice University (2007).

[24] Y. Wang, J. Yang, W. Yin, Y. Zhang, A new alternating minimization algorithm for total variation image reconstruction, SIAM J. Img. Sci. 1 (2008) 248–272.

[25] C. Wu, J. Zhang, X.-C. Tai, Augmented Lagrangian method for total variation restoration with non-quadratic fidelity, Inverse Problems and Imaging 5 (2011) 237–261.

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

[27] J.-L. Starck, M. Elad, D. L. Donoho, Image decomposition: Separation of texture from piecewise smooth content, in: Storage and Retrieval for Image and Video Databases, 2003.

[28] R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010.

[29] A. Delaunoy, K. Fundana, E. Prados, A. Heyden, Convex multi-region segmentation on manifolds, in: The 12th IEEE International Conference on Computer Vision, IEEE, 2009, pp. 662 – 669.

[30] C. Wu, J. Zhang, Y. Duan, X.-C. Tai, Augmented Lagrangian method for total variation based image restoration and segmentation over triangulated surfaces, Journal of Scientific Computing 50 (2012) 145–166.

[31] M. Bertalmío, L.-T. Cheng, S. Osher, G. Sapiro, Variational problems and partial differential equations on implicit surfaces, J. Comput. Phys. 174 (2001) 759–780.

[32] M. Krueger, P. Delmas, G. Gimel'Farb, Active contour based segmentation of 3d surfaces, in: Proceedings of the 10th European Conference on Computer Vision: Part II, ECCV '08, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 350–363.

[33] J. Stam, Flows on surfaces of arbitrary topology, ACM Trans. Graph. 22 (2003) 724–731.

[34] X. Gu, S.-T. Yau, Global conformal surface parameterization, in: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing, SGP '03, Eurographics Association, Switzerland, 2003, pp. 127–137.

[35] L. M. Lui, X. Gu, T. F. Chan, S.-T. Yau, Variational method on riemann surfaces using conformal parameterization and its applications to image processing, Methods and Applications of Analysis 15 (2008) 513–538.

[36] C. Wu, J. Deng, F. Chen, Diffusion equations over arbitrary triangulated surfaces for filtering and texture applications, IEEE Transactions on Visualization and Computer Graphics 14 (2008) 666–679.

[37] C. Wu, J. Deng, F. Chen, X. Tai, Scale-space analysis of discrete filtering over arbitrary triangulated surfaces, SIAM J. Img. Sci. 2 (2009) 670–709.

[38] R. Lai, T. F. Chan, A framework for intrinsic image processing on surfaces, Tech. rep., Department of Mathematics, UCLA, Los Angeles, CA 90095, USA (2010).

[39] C. Wu, X.-C. Tai, Augmented Lagrangian method, dual methods, and split bregman iteration for rof, vectorial tv, and high order models, SIAM J. Img. Sci. 3 (2010) 300–339.

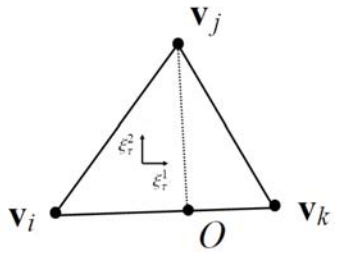

Figure 1: Local parameterization of a triangle.

Table 1: Statistics of the 3D meshes used in the experiments.

Mesh surface	No. of vertices	No. of triangles
Lena (Fig. 5)	65536	130050
Fingerprint (Fig. 4)	65536	130050
Barbara (Fig. 2)	262144	522242
Cup (Fig. 6)	131584	262146
Bunny 1 (Fig. 3)	34834	69451
Bunny 2 (Fig. 7)	34817	69630

(a) Input (b) L^2, $\mathbf{u}$ (c) L^2, $\mathbf{v}$ (d) L^1, $\mathbf{u}$ (e) L^1, $\mathbf{v}$

Figure 2: The top row is conventional L^2 and L^1 image decomposition on 2D plane. The bottom row is our proposed decomposition results on planar manifold. (b) and (d) are the structures. (c) and (e) are the textures.

(a) Textured bunny 1 (b) L^2, $\mathbf{u}$ (c) L^2, $\mathbf{v}$ (d) L^1, $\mathbf{u}$ (e) L^1, $\mathbf{v}$

Figure 3: Proposed image decomposition on a triangular surface. (a) is the input textured bunny. (b) and (d) are the extracted structures extracted using L^2 and L^1 norms, respectively. (c) and (e) are the extracted textural parts using L^2 and L^1 norms, respectively.

Table 2: Statistics for Fig. 2 and Fig. 3, where N is the iteration number and t is the running time. The stop condition is $\epsilon_0 = 1.e^{-6}$.

Models	L^2 norm				L^1 norm				
Parameters	λ	r_p	t	N	λ	r_p	r_z	t	N
Barbara (plane)	5.00	15	12.144s	12	0.30	15	5	17.775s	17
Barbara (manifold)	400	0.015	14.865s	16	40	0.015	200	15.354s	17
Bunny 1	200	0.02	4.279s	22	60	0.02	200	8.298s	43

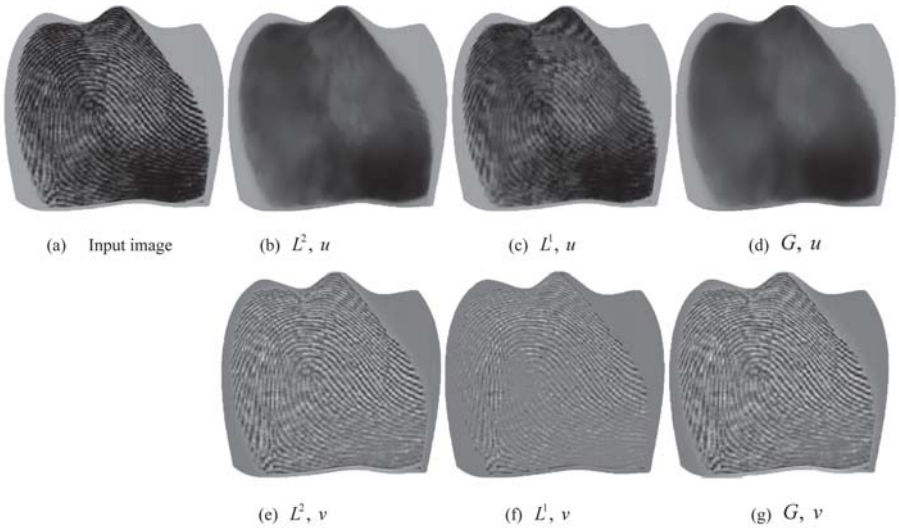

Figure 4: Decomposition of an image with oscillatory pattern by the proposed method. (a) is the input image with oscillatory pattern. (b), (c) and (d) are the extracted structural components using L^2, L^1 and G norms, respectively. (e), (f) and (g) are the extracted textural components using L^2, L^1 and G norms, respectively.

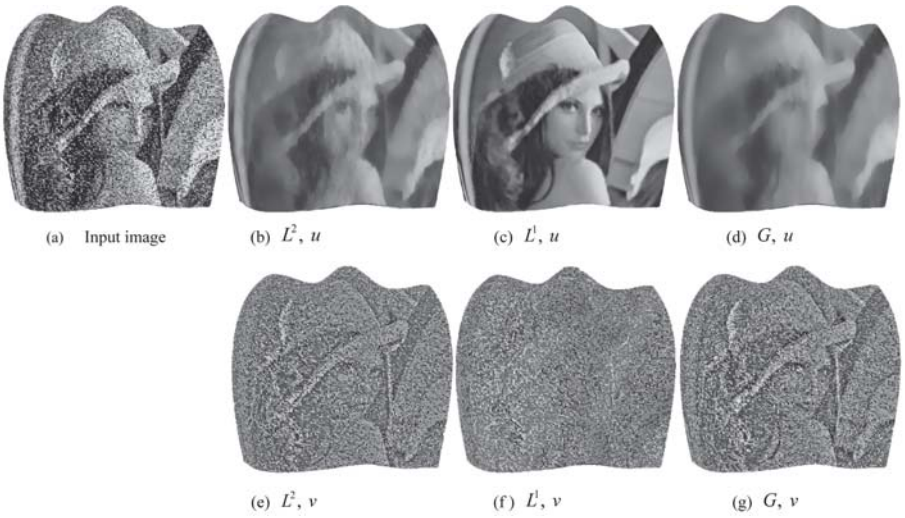

Figure 5: Decomposition of an image with sparse textures by the proposed method. (a) is the input image with salt and pepper noise. (b) and (c) are the structural image and noise extracted from (a) using L^2 norm. (d) and (e) are the structural image and noise extracted from (a) using L^1 norm. (f) and (g) are the structural image and noise extracted from (a) using G norm.

(a) Textured cup (b) L^2, $\mathbf{u}$ (c) L^2 exaggerated detail (d) L^1, $\mathbf{u}$ (e) L^1 exaggerated detail

Figure 6: Detail exaggeration. (a) is the textured cup, (b) and (d) are the extracted structural components using L^2 and L^1 norms. (c) and (e) are the exaggerated detail results by simply scaling the texture components. L^2 norm: $\lambda = 300, r_p = 0.012, t = 10.659s, N = 15$; L^1 norm: $\lambda = 100, r_p = 0.02, r_z = 200, t = 11.225.s, N = 16$.

(a) 10%, gray (b) 20%, gray (c) 10%, color (d) 20%, color

Figure 7: Comparison of the proposed method and the L^2 method of [30] for denoising bunny 2. The models in the first row are gray and color textured manifolds added with $10\%, 20\%$ salt and pepper noise, respectively. The models in the second row are the denoising results using the method of [30]. The models in the third row are the denoising results using the proposed TV-L^1 method.

Table 3: Statistics for Fig. 7 where N is the iteration number and t is the running time.

Models		L^2				
Parameters	SNR (dB)	λ	r_p	N	t	SNR(dB)
Bunny (10% gray)	-5.740	300	0.015	22	1.934s	6.523
Bunny (20% gray)	-8.560	280	0.015	24	1.948s	1.806
Bunny (10% color)	-2.789	300	0.015	26	4.320s	9.038
Bunny (20% color)	-5.637	250	0.015	28	4.439s	4.639
Models		L^1				
Parameters	SNR (dB)	λ	r_p	r_z	N	t	SNR(dB)
Bunny (10% gray)	-5.740	120	0.015	200	19	1.717s	13.304
Bunny (20% gray)	-8.560	120	0.015	200	22	1.988s	10.847
Bunny (10% color)	-2.789	120	0.015	200	21	4.115s	15.921
Bunny (20% color)	-5.637	120	0.015	200	23	4.224s	12.075