



HAL
open science

Convex Multi-Region Segmentation on Manifolds

Amael Delaunoy, Ketut Fundana, Emmanuel Prados, Anders Heyden

► **To cite this version:**

Amael Delaunoy, Ketut Fundana, Emmanuel Prados, Anders Heyden. Convex Multi-Region Segmentation on Manifolds. ICCV 2009 - 12th IEEE International Conference on Computer Vision, Sep 2009, Kyoto, Japan. pp.662-669, 10.1109/ICCV.2009.5459174 . inria-00435133

HAL Id: inria-00435133

<https://inria.hal.science/inria-00435133v1>

Submitted on 23 Nov 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Convex Multi-Region Segmentation on Manifolds

Amaël Delaunoy¹ Ketut Fundana² Emmanuel Prados¹ Anders Heyden³

¹ INRIA Rhône-Alpes / LJK, Grenoble, France

² Applied Mathematics Group, Malmö University, Sweden
and Image Group, Copenhagen University, Denmark

³ Mathematical Imaging Group, Lund University of Technology, Sweden

Abstract

In this paper, we address the problem of segmenting data defined on a manifold into a set of regions with uniform properties. In particular, we propose a numerical method when the manifold is represented by a triangular mesh. Based on recent image segmentation models, our method minimizes a convex energy and then enjoys significant favorable properties: it is robust to initialization and avoid the problem of the existence of local minima present in many variational models. The contributions of this paper are threefold: firstly we adapt the convex image labeling model to manifolds; in particular the total variation formulation. Secondly we show how to implement the proposed method on triangular meshes, and finally we show how to use and combine the method in other computer vision problems, such as 3D reconstruction. We demonstrate the efficiency of our method by testing it on various data.

1. Introduction

Image segmentation aims to partition a given image into several meaningful regions based on certain attributes such as intensity, texture, color, etc. This problem is one of the most challenging and important problems in computer vision. We address the problem of segmenting *data* defined on *manifolds* (typically a 2-surface in \mathbb{R}^3) into multiple regions of piecewise constant attributes. The ability to solve such a problem offers significant new possibilities in a number of applications. For example, in 3D reconstruction (see Jin et al. [18]), a segmentation into piecewise constant data of the reconstructed surface allows to naturally introduce constraints on the material of the scene.

1.1. Global Multi-Region Segmentation

Many approaches have been proposed to solve image segmentation problems. In particular, via gradient descents,

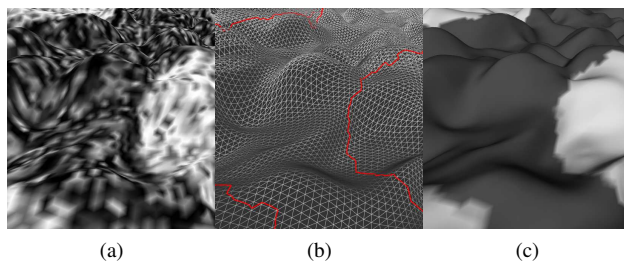


Figure 1: Example of segmentation on a manifold. (a) The input textured surface. (b) The triangular representation of the surface with the retrieved contour (in red). (c) Surface colored with the mean values of the segmented regions (and surface shading).

variational methods for image segmentation have had a great success, such as snakes [20], geodesic active contours [8], geodesic active region [29] and the Chan-Vese models [10]. Yet, the main drawback of those methods is the existence of local minima due to the non-convexity of the energy functionals. Minimizing those functionals by gradient descent methods makes the initialization critical.

To obtain global minima, some previous image segmentation works have used different optimization techniques: For example the graph-cuts in a fully discrete setting, see [6, 21, 22] and the references therein. Nevertheless, while binary segmentation methods based on graph-cuts assure to get a global minima, multi-region segmentation algorithms are based on sequences of graph-cuts which cannot guarantee a global optimization.

Recently, some authors have tried to handle the problem in another direction. Instead of working on the optimization techniques in order to compute the minima of non-convex problems, they have reformulated the energy in order to get a fully convex problem [2, 7, 9, 11, 24, 31, 35]. These segmentation techniques are based on TV-regularizers and aim at finding characteristic functions that minimize the objective functions. Obtaining *global minima* becomes easy and can be done by simply performing a gradient descent. Also

the *initialization problem vanishes*: the algorithm can start from any initialization and obtains the same result. The multi-region segmentation models proposed by [9, 24, 35] are rather similar, the work of Pock et al. [31] (inspired from Ishikawa’s [17]) differs from the fact that it deals with ordered labels and uses a regularization term which favors transitions between nearby labels. This makes sense in their stereo application where the ordering is due to depth, but it is not the case when we deal with independent labels. Here, we adopt the model of [9, 24, 35] which is more appropriate to the applications we have in mind (the fact that real-life scenes are made by a finite number of independent materials, in 3D reconstruction problems) and we adapt this *image* labeling model to *manifolds*.

1.2. Data Segmentation on Manifolds

Manifolds such as surfaces are common in computer graphics as well as computer vision. Although data segmentation on surfaces has been recently used on implicit representations (see for example [18, 23]), explicit representations such as triangular meshes are natural and intuitive representations. Mesh representations have been widely used in 3D reconstruction, see for example [1, 12, 33], and recently in [1, 32, 36], which takes advantage of various recent evolution methods. These allow us to naturally deal with topological changes (necessary property e.g. in 3D shape estimation problems formulated within a variational framework). In a number of applications, particularly in graphics, this is the only representation one has at ones disposal. In this paper, after adapting the image segmentation model to manifolds, we show how to implement the proposed method on *triangular meshes*.

Mesh segmentation has been used in computer graphics to decompose meshes into significant parts, but previous work mainly focus on the geometric aspects, the choice and the representations of features to use (like the curvature). We refer to [3] for a recent survey of those techniques, as in this work we focus in segmenting data on the mesh.

However, the problem of segmenting data like texture on manifolds has not received much attention until now, and is quite different from the geometric decomposition of a mesh. In [30], authors take into account both texture information and curvature, but their approach is based on a fast marching algorithm, which needs to be initialized using initial points. Moreover as their segmentation method is not convex, different regions sharing the same properties may result in different labels. The same problem also occurs in [28], where watershed filtering was used. Contrary to those methods, the approach we propose is global and robust to initialization.

Finally, let us note that the segmentation model we consider is based on a total variation regularization. Although this regularization has been previously used on implicit sur-

faces (see [4] with applications to texture synthesis), it has not received much attention in Lagrangian methods. In the finite elements literature, one can find some papers dealing with the Laplace-Beltrami operator $\nabla_{\mathcal{S}} \cdot \nabla_{\mathcal{S}} u$ which corresponds to the gradient of the squared regularization term $\int_{\mathcal{S}} |\nabla_{\mathcal{S}} u|^2 d\sigma$ (see for example [13, 15]). To our knowledge there does not exist work dealing with the gradient of the total variation in this framework. Let us note here that the Laplace-Beltrami operator is linear while the term associated with the gradient of the total variation $\nabla_{\mathcal{S}} \cdot \frac{\nabla_{\mathcal{S}} u}{|\nabla_{\mathcal{S}} u|}$ is nonlinear. Moreover, the theoretical analysis and the numerical algorithms of [13, 15] are completely based on this linearity property.

1.3. Contributions

First, we adapt the image convex model of [7, 9, 11, 24, 35] to *manifolds*. Then we show how to implement the method when the manifold is represented by a triangular *mesh*. Finally, we explain how our multi-region segmentation method could be incorporated into potential computer vision applications such as *3D reconstruction*.

2. Multi-Region Segmentation Model

In this section, we describe the convex image segmentation model we propose. To make this model comprehensible and intuitive, let us first remind of the region-based active contour model of Chan and Vese [10]. Here we show that the energy functional of Chan and Vese, which is the piecewise constant case of the Mumford-Shah model ([27]), can be recast as a convex functional in order to find the global minimizer of the original energy functional.

2.1. Convex Two-Phases Model

The Chan-Vese model [10], which is formulated in the level set framework, partition a given image into two sub-regions. For a given image I , the idea is to find a subset Σ of a bounded domain $\Omega \subset \mathbb{R}^N$, whose boundary $\partial\Sigma$ is represented by the zero level set of function $\phi : \Omega \rightarrow \mathbb{R}^N$. This is done by minimizing the energy functional

$$\min_{\phi, c_1, c_2} \int_{\Omega} \left\{ H_{\epsilon}(\phi)(I(\mathbf{w}) - c_1)^2 + (1 - H_{\epsilon}(\phi))(I(\mathbf{w}) - c_2)^2 + \lambda |\nabla H_{\epsilon}(\phi)| \right\} d\omega, \quad (1)$$

where $\lambda \in \mathbb{R}$, $c_1, c_2 \in \mathbb{R}$ and H_{ϵ} is a regularized Heaviside function, which models a characteristic function (see [10]).

Since the energy functional (1) is not convex, minimizing it by gradient descent methods can get stuck in local minima. By relaxing the characteristic function $H_{\epsilon}(\phi)$ by an arbitrary function u bounded between 0 and 1, Chan et

al. [11] showed that minimizing (1) can be rewritten as the following convex minimization problem:

$$\min_{0 \leq u \leq 1} \left\{ \int_{\Omega} \left\{ u(\mathbf{w})(I(\mathbf{w}) - c_1)^2 + (1 - u(\mathbf{w}))(I(\mathbf{w}) - c_2)^2 \right\} d\mathbf{w} + \lambda \int_{\Omega} |\nabla u| d\mathbf{w} \right\}, \quad (2)$$

c_1 and c_2 being fixed, in \mathbb{R} . As proved in [7, 11], if $u(x)$ is a minimizer of (2), then for a.e. $\mu \in [0, 1]$, the set $\Sigma(\mu) = \{x \in \Omega, u(x) > \mu\}$ is a minimizer of the Mumford-Shah functional [27], implying that *the solution to (1) can be obtained by thresholding u at any arbitrary threshold between 0 and 1.*

2.2. Extension to Multi-Region Segmentation

Recently, several authors [9, 24, 35] have extended the convex formulation (2) to multi-region segmentation:

$$\min_{\mathbf{u} \in K} \left\{ \int_{\Omega} \langle \mathbf{u}(\mathbf{w}), \mathbf{s}(\mathbf{w}) \rangle + \lambda |\nabla \mathbf{u}(\mathbf{w})| d\mathbf{w} \right\}, \quad (3)$$

where K is the set of function $\mathbf{u} : \Omega \rightarrow \mathbb{R}^m$ such that for all $\mathbf{w} \in \Omega$ and $p \in [1..m]$, $\mathbf{u}_p(\mathbf{w}) \geq 0$ and $\sum_{p=1}^m \mathbf{u}_p(\mathbf{w}) = 1$. $|\nabla \mathbf{u}(\mathbf{w})|$ corresponds to $\sqrt{\sum_p |\nabla \mathbf{u}_p(\mathbf{w})|^2}$, where $|\cdot|$ denotes the L^2 norm. m denotes the number of labels and $\mathbf{s}(\mathbf{w})$ is an m -dimensional vector; $s_p(\mathbf{w})$ indicates the affinity of the data at point \mathbf{w} with class p . The convex domain naturally allow direct competition between the labeling.

3. Multi-Region Segmentation on Manifolds

In this section we extend the multi-region convex model (3) on a manifold, and we show how to optimize the associated energy for a manifold represented by a mesh. To our best knowledge, these convex formulations (2,3) have been defined only on open subsets of \mathbb{R}^N which correspond to image domains, as described in the previous section.

Let \mathcal{S} be a Riemannian manifold. Typically, \mathcal{S} could be a smooth 2D surface of \mathbb{R}^3 . Energy (3) is adapted as follows:

$$\min_{\mathbf{u} \in K} \left\{ \int_{\mathcal{S}} \langle \mathbf{u}(\mathbf{x}), \mathbf{s}(\mathbf{x}) \rangle + \lambda |\nabla_{\mathcal{S}} \mathbf{u}(\mathbf{x})| d\sigma \right\}, \quad (4)$$

where now the functions \mathbf{u} are defined on \mathcal{S} instead of Ω , $|\cdot|$ is the Riemannian norm, $\nabla_{\mathcal{S}}$ is the intrinsic gradient on \mathcal{S} and $d\sigma$ is the manifold's element measure (surface's area measure for 2D manifolds).

Now, let us consider a manifold represented by a mesh. The following results apply to manifolds with any topology. Let \mathbf{X} be a (piecewise linear) polyhedron representation of the surface \mathcal{S} , defined by a set of vertices $\mathbf{x}_k : \mathbf{X} = \{\mathbf{x}_k\}$ and let l be the cardinality of \mathbf{X} (the number of vertices). As in the finite elements literature, we define $\phi_k : \mathcal{S} \rightarrow \mathbb{R}$ as

the piecewise affine, interpolating basis function such that $\phi_k(\mathbf{x}_k) = 1$ and $\phi_k(\mathbf{x}_i) = 0$ if $i \neq k$. The vector valued field $\mathbf{U} = \{\mathbf{u}_k\}$ is defined on all vertices \mathbf{x} of the polyhedron \mathbf{X} . \mathbf{U} can be naturally extended on \mathcal{S} by a piecewise affine vector valued field on \mathcal{S} . We denote this extension $\mathbf{u}(\mathbf{x}) = \sum_k \mathbf{u}_k \phi_k(\mathbf{x})$. To make the paper easier to read and because of space limitations, we assume that the manifold is a 2D surface of \mathbb{R}^3 . However, the following method applies to any dimension. Let \mathcal{S}_j be the j^{th} triangle of the mesh. The multi-region segmentation energy can then be rewritten as

$$\sum_j \sum_k \left\langle \mathbf{u}_k, \int_{\mathcal{S}_j} \phi_k(\mathbf{x}) \mathbf{s}(\mathbf{x}) d\sigma \right\rangle + \lambda \int_{\mathcal{S}_j} |\nabla_{\mathcal{S}} \mathbf{u}(\mathbf{x})| d\sigma, \quad (5)$$

where \mathbf{u} is constrained to be in K . The first term of (5) is explicitly written with respect to \mathbf{U} . In order to make the total variation term explicit with respect to \mathbf{U} , we first consider a local parametrization (α, β) on the manifold. Following [14, 19], we rewrite the right term of Equation (5) using fundamental forms:

$$\nabla_{\mathcal{S}} \mathbf{u} = \begin{bmatrix} \frac{\partial \mathbf{x}}{\partial \alpha} & \frac{\partial \mathbf{x}}{\partial \beta} \end{bmatrix} \begin{bmatrix} E & F \\ F & G \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{u}_{\alpha} \\ \mathbf{u}_{\beta} \end{bmatrix}, \text{ and then}$$

$$|\nabla_{\mathcal{S}} \mathbf{u}| = \sqrt{\begin{bmatrix} \mathbf{u}_{\alpha} & \mathbf{u}_{\beta} \end{bmatrix} \begin{bmatrix} E & F \\ F & G \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{u}_{\alpha} \\ \mathbf{u}_{\beta} \end{bmatrix}},$$

where $E = \frac{\partial \mathbf{x}}{\partial \alpha} \cdot \frac{\partial \mathbf{x}}{\partial \alpha}$, $F = \frac{\partial \mathbf{x}}{\partial \alpha} \cdot \frac{\partial \mathbf{x}}{\partial \beta}$ and $G = \frac{\partial \mathbf{x}}{\partial \beta} \cdot \frac{\partial \mathbf{x}}{\partial \beta}$ are coefficients of the first fundamental form (see [14, 19]). \mathbf{u}_{α} and \mathbf{u}_{β} are partial derivatives of \mathbf{u} with respects to α and β respectively. Considering the mesh representation, we parametrize the triangle \mathcal{S}_j by $\mathbf{x}(\alpha, \beta) = \mathbf{x}_{j,1} + \alpha \overrightarrow{\mathbf{x}_{j,1}\mathbf{x}_{j,2}} + \beta \overrightarrow{\mathbf{x}_{j,1}\mathbf{x}_{j,3}}$ where $\mathbf{x}_{j,1}$, $\mathbf{x}_{j,2}$ and $\mathbf{x}_{j,3}$ are the three vertices associated with the triangle \mathcal{S}_j and where $(\alpha, \beta) \in T = \{(\alpha, \beta) | \alpha \in [0, 1] \text{ and } \beta \in [0, 1 - \alpha]\}$. We then have $\int_{\mathcal{S}_j} |\nabla_{\mathcal{S}} \mathbf{u}(\mathbf{x})| d\sigma =$

$$\int_T \sqrt{\sum_p \mathbf{u}_{p\alpha}^2 G - 2\mathbf{u}_{p\alpha} \cdot \mathbf{u}_{p\beta} F + \mathbf{u}_{p\beta}^2 E} d\alpha d\beta. \quad (6)$$

$\mathbf{u}_{p\alpha}$ and $\mathbf{u}_{p\beta}$ are partial derivatives of \mathbf{u}_p with respects to α and β respectively. Here the reader will easily verify that $E, F, G, \mathbf{u}_{p\alpha}$ and $\mathbf{u}_{p\beta}$ are constant functions on \mathcal{S}_j and that their respective values are equal to $E_j = |\mathbf{x}_{j,2} - \mathbf{x}_{j,1}|^2$, $F_j = \langle \mathbf{x}_{j,2} - \mathbf{x}_{j,1}, \mathbf{x}_{j,3} - \mathbf{x}_{j,1} \rangle$, $G_j = |\mathbf{x}_{j,3} - \mathbf{x}_{j,1}|^2$, $\mathbf{u}_{p\alpha}^j = \mathbf{u}_{j,2p} - \mathbf{u}_{j,1p}$ and $\mathbf{u}_{p\beta}^j = \mathbf{u}_{j,3p} - \mathbf{u}_{j,1p}$, where $\mathbf{u}_{j,1}$, $\mathbf{u}_{j,2}$ and $\mathbf{u}_{j,3}$ are the values of \mathbf{u} at vertices $\mathbf{x}_{j,1}$, $\mathbf{x}_{j,2}$ and $\mathbf{x}_{j,3}$ respectively. Now the term inside the integral of (6) does not depend on α and β . The convex multi-region

segmentation energy on the meshed manifold becomes:

$$E(\mathbf{U}) = \sum_j \sum_k \left\langle \mathbf{u}_k, \int_{S_j} \phi_k(\mathbf{x}) s(\mathbf{x}) d\sigma \right\rangle + \frac{\lambda}{2} \sum_j \sqrt{\sum_p \mathbf{u}_{p\alpha}^j{}^2 G_j - 2\mathbf{u}_{p\alpha}^j \cdot \mathbf{u}_{p\beta}^j F_j + \mathbf{u}_{p\beta}^j{}^2 E_j}. \quad (7)$$

3.1. Optimization Method

When the (surface) manifold is represented by a mesh, the convex multi-region segmentation model then leads to optimizing the convex energy (7) with respect to $\mathbf{U} \in \mathbb{R}^{l \times m}$, with the convex constraint $\mathbf{U} \in K$; K being the set $\{\mathbf{U} \text{ s.t. } \forall k, \sum_p \mathbf{u}_{kp} = 1 \text{ and } \forall p, \mathbf{u}_{kp} \geq 0\}$. This convex constrained optimization problem on $\mathbb{R}^{l \times m}$ can be solved by the projected gradient method [5], which consists in generating the sequence U^t via

$$\mathbf{U}^{t+1} = Proj_K(\mathbf{U}^t - \tau \nabla E(\mathbf{U}^t)), \quad (8)$$

for a fixed time step $\tau > 0$, until $|\mathbf{U}^t - \mathbf{U}^{t-1}|_\infty \leq \delta$, a small constant. $Proj_K$ is the projection on the convex set K . In other words, we iteratively process gradient descent steps and projections of the \mathbf{u}_k on the set K . These projections can be done via Michelot's algorithm [26]. From energy (7) we easily obtain

$$\frac{\partial E}{\partial \mathbf{u}_{kp}}(\mathbf{U}) = \sum_{j \in N(k)} \left[\int_{S_j} \phi_k(\mathbf{x}) s(\mathbf{x}) d\sigma \right]_p - \frac{\lambda}{2} Q(\xi + \epsilon)^{-\frac{1}{2}}, \quad (9)$$

where $Q = (\mathbf{u}_{j,2p} - \mathbf{u}_{kp})(G_j - F_j) + (\mathbf{u}_{j,3p} - \mathbf{u}_{kp})(E_j - F_j)$, ξ is the term in the squared root of (7), and $N(k)$ is the 1-ring neighborhood of vertex k . As in [11], we regularize the term ξ by incorporating a small value ϵ inside the squared root to avoid instabilities when the gradient of \mathbf{u} is 0.

Let us remind now that, as underlined by [16], the notation of gradient depends on the underlying scalar product. If we chose the pointwise scalar product $\langle \mathbf{U}, \mathbf{V} \rangle_{pw} = \sum_k \langle \mathbf{u}_k, \mathbf{v}_k \rangle$, then the components of $\nabla E(\mathbf{U}^t)$ directly coincide with $\frac{\partial E}{\partial \mathbf{u}_{kp}}(\mathbf{U})$. Nevertheless the associate pointwise metric is not efficient for minimizing energies of the form $\int_S f(\mathbf{u}(\mathbf{x})) d\sigma$ since the distance between two discrete fields \mathbf{U} and \mathbf{V} does not take into account the area of the triangle. On the other hand, the L^2 scalar product $\langle \mathbf{U}, \mathbf{V} \rangle_{L^2} = \int_S \langle \mathbf{u}(\mathbf{x}), \mathbf{v}(\mathbf{x}) \rangle d\sigma$ is much more relevant. Also in this case the gradient becomes

$$\nabla E(\mathbf{U}) = M^{-1} \frac{\partial E}{\partial \mathbf{U}}(\mathbf{U}), \quad (10)$$

where the matrix M is the mass matrix defined by $M_{ij} = Id_m \int_S \phi_i(x) \phi_j(x) d\sigma$. Moreover one classically approximates M by the diagonal mass lumping \tilde{M} , where \tilde{M}_{ii} is the area of the Voronoi dual cell of \mathbf{x}_i times the identity matrix Id_m , see e.g. [16].

3.2. Applications

In the previous sections the data term of the segmentation model s is assumed to be known (Equation 4). In the applications, this term also depends on some parameters that have to be optimized. The convex problem can be solved by alternating optimization of the parameters in a bi-convex way. For fixed parameters of s we update \mathbf{u} and vice-versa. \mathbf{u} is updated according to the method presented in section 3.1. In practice we update the parameters of s every r update iterations of \mathbf{u} (r is chosen arbitrary; we fix $r = 10$ in our experiments).

Piecewise Constant Data Segmentation

Let us consider the case where the data we want to segment are assumed to be piecewise constant. Here a natural expression for $s_p(\mathbf{x})$ is to use the squared error between the scalar or vector-valued data $C(\mathbf{x})$ at the point \mathbf{x} and the value μ_p associated with the label p (μ_p having the same dimension as the data):

$$s_p(\mathbf{x}) = (C(\mathbf{x}) - \mu_p)^T (C(\mathbf{x}) - \mu_p).$$

The optimization of the energy (4) with respect to μ_p gives:

$$\mu_p = \frac{\int_S \mathbf{u}_p(\mathbf{x}) C(\mathbf{x}) d\sigma}{\int_S \mathbf{u}_p(\mathbf{x}) d\sigma},$$

which corresponds to the mean value of the data of the associated region. Note that the previous model can be easily extended to any probability density function D_p . For example, D_p can be a multivariate gaussian density function of mean μ_p and covariance Σ_p , and then we would have:

$$s_p(\mathbf{x}) = -\ln(D_p(\mathbf{x}, \mu_p, \Sigma_p)), \text{ with}$$

$$D_p(\mathbf{x}, \mu_p, \Sigma_p) = \frac{1}{\sqrt{2\pi}^{|\Sigma|}} e^{-\frac{1}{2}(C(\mathbf{x}) - \mu_p)^T \Sigma_p^{-1} (C(\mathbf{x}) - \mu_p)}.$$

Segmentation in 3D Reconstruction Problems

Such segmentation framework can be incorporated in 3D Reconstruction applications. In such applications, it can be interesting to segment a particular region, or all parts of the surface sharing the same reflectance properties. In 3D reconstruction, most of the variational methods yield to minimizing an energy of the form

$$E(S) = \sum_i \int_S g(\mathbf{x}) \frac{\mathbf{x}_i \cdot \mathbf{n}}{\mathbf{x}_{i,z}^3} \nu_S(\mathbf{x}) d\sigma, \quad (11)$$

see for example [12, 18, 19]. Moreover, if we choose

$$g(\mathbf{x}) = \sum_{p=1}^m \mathbf{u}_p(\mathbf{x}) (I_i(\pi_i(\mathbf{x})) - \mu_p)^T (I_i(\pi_i(\mathbf{x})) - \mu_p),$$

where $\pi_i(\mathbf{x})$ is the projection of the surface point \mathbf{x} into the i^{th} image and $I_i : w \mapsto I_i(w)$ is the function which associates to each pixel w , its color on the i^{th} image. We then get an extension of the stereoscopic segmentation method proposed by [34] to the case where the surface is composed of more than two regions of piecewise constant radiance. Also, contrary to our method, the segmentation approach proposed in [34] is subject to local minima. Finally, the optimization of the energy (4) with respect to μ_p gives:

$$\mu_p = \frac{\int_S \mathbf{u}_p(\mathbf{x}) \sum_i I_i(\pi_i(\mathbf{x})) \frac{\mathbf{x} \cdot \mathbf{n}}{\mathbf{x}_z^3} \nu_{S,i}(\mathbf{x}) d\sigma}{\int_S \mathbf{u}_p(\mathbf{x}) \sum_i \frac{\mathbf{x} \cdot \mathbf{n}}{\mathbf{x}_z^3} \nu_{S,i}(\mathbf{x}) d\sigma}.$$

If we chose $g(\mathbf{x}) =$

$$\sum_{p=1}^m \mathbf{u}_p(\mathbf{x}) (I_i(\pi_i(\mathbf{x})) - \rho_p \mathbf{N}(\mathbf{x}) \cdot \mathbf{L})^T (I_i(\pi_i(\mathbf{x})) - \rho_p \mathbf{N}(\mathbf{x}) \cdot \mathbf{L}),$$

where $\mathbf{N}(\mathbf{x})$ is the normal to the surface at the point \mathbf{x} and \mathbf{L} is the vector corresponding to the light source illuminating the scene, then we get an extension of the (Lambertian) multi-view shape from shading method proposed by [18] for surfaces with piecewise constant albedo. In the same way, contrary to our approach, the method proposed by [18] is limited to two regions segmentation and is strongly subject to local minima. The optimization of energy (4) with respect to the albedo gives:

$$\rho_p = \frac{\int_S \mathbf{u}_p(\mathbf{x}) \sum_i I_i(\pi_i(\mathbf{x})) \mathbf{N}(\mathbf{x}) \cdot \mathbf{L} \frac{\mathbf{x} \cdot \mathbf{n}}{\mathbf{x}_z^3} \nu_{S,i}(\mathbf{x}) d\sigma}{\int_S \mathbf{u}_p(\mathbf{x}) (\mathbf{N}(\mathbf{x}) \cdot \mathbf{L})^2 \sum_i \frac{\mathbf{x} \cdot \mathbf{n}}{\mathbf{x}_z^3} \nu_{S,i}(\mathbf{x}) d\sigma}.$$

The theoretical and experimental study of these algorithms will be the topic a forthcoming paper.

4. Experiments

In order to validate the proposed multi-region segmentation approach on meshes, we present different experiments on synthetic as well as realistic data. In practice as explained in previous section, the segmentation is solved by alternating between region parameters and the segmentation variable \mathbf{U} , with a known number of regions. The algorithm complexity is linearly dependent on the number of facets and the number of classes. Experiments have been runned on a 2.66GHz linux machine and take about 20 seconds on a mesh of 200,000 facets and for a 4 regions segmentation. The values of λ have been manually chosen in each example but a value of 0.01 gives reasonable results in most cases.

4.1. The Two Region Case

Figures 2 and 3 show examples of our algorithm using a synthetic image mapped onto a mesh for the Stanford bunny model. Noise has been added to the image. Here, we show that our algorithm performs well on the given example and that the final solution is binary. Moreover it is

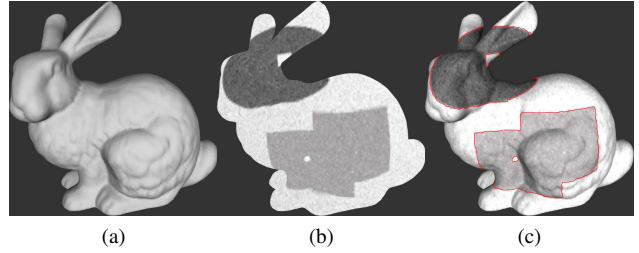


Figure 2: Segmentation result on the synthesized Stanford bunny surface. (a) Input shaded object. (b) Input mesh with synthetic texture mapping. (c) Input textured mesh (shaded visualization) and final contour (in red).

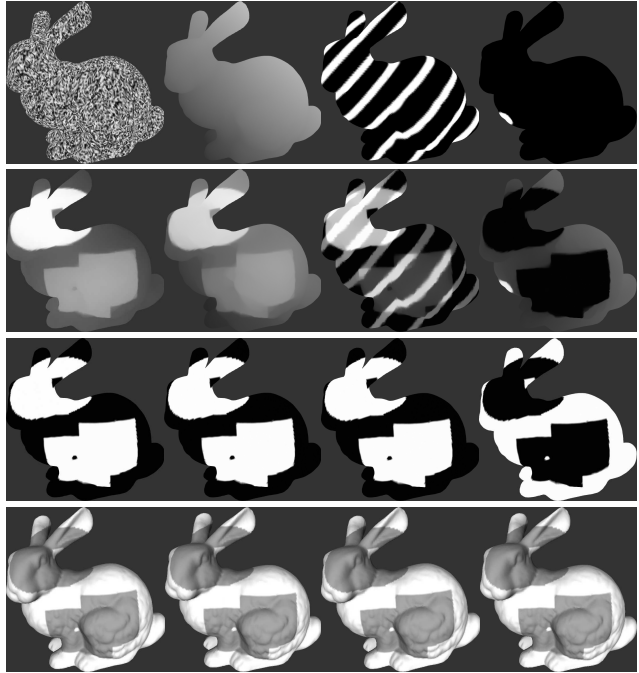


Figure 3: The evolution on the synthesized bunny surface. Different initialization of \mathbf{U} (first row); Intermediate values of \mathbf{U} (second row); The obtained solution \mathbf{U} (third row); The obtained mean values (fourth row) with shading.

robust to the initialization of the scalar function \mathbf{U} . Note that the retrieved solution that has been displayed is the auxiliary value \mathbf{u} , and not the segmented constant values μ_1 and μ_2 . Also because the energy functional is convex in \mathbf{u} only and the values μ_1 and μ_2 are optimized during the evolution, they can be assigned to the region corresponding to either $\mathbf{u} = (1, 0)$ or $\mathbf{u} = (0, 1)$, this explains why the last initialization do not show the same values of \mathbf{u} but an inverted one. In practice, although the total functional is not fully convex, we obtain the same results and really similar μ_1 and μ_2 for each example. We respectively obtain $(\mu_1 = 140.778, \mu_2 = 231.003)$, $(\mu_1 = 140.746, \mu_2 = 231.01)$, $(\mu_1 = 140.75, \mu_2 = 231.03)$ and $(\mu_1 = 230.992, \mu_2 = 140.765)$ for the four different initializations. Note that in the last column, values of μ_1

and μ_2 are inverted and the solution \mathbf{u} as well. In this example, geodesic active contours or level sets methods would tend to the closest local minima from the initialization as the texture is not clearly binary. Nevertheless, as the method is global here, segmenting a particular region should be done using additional cues.

Figure 4 present segmented surfaces from real-world textures that have been mapped onto a mesh, in the case of the two-phases segmentation. We show different examples from classical images used in segmentation. Note that the segmentation is done on the mesh using the method described in this paper and not on an image. The experiments show three different non binary images and their segmentation into two different regions. As expected the results are binary even though the initial values of the segments are random values. The mean values of each region is es-



Figure 4: Segmentation results on meshes in the two-region segmentation on three different examples. From top to bottom: Input textured mesh; Mesh shape where the segmentation is performed and the initial random value of one component of \mathbf{U} ; Recovered mean values of each region; Segmented object.

timated during the process as described before, and the parameters λ can be adjusted to add more smoothness to the segmentation. As shown by experiments, even though the initialization is random and the parameters of each region are computed during the evolution, the algorithm still converges to the desired solution as a binary solution.

4.2. Dealing with Multiple Regions

Here we show the efficiency of the proposed method when dealing with multiple regions. Different examples are shown, first with synthetic textures on which noise has been added, and then on meshes textured by real-world images like previous examples. Note that the number of regions is initially given and is not automatically estimated.

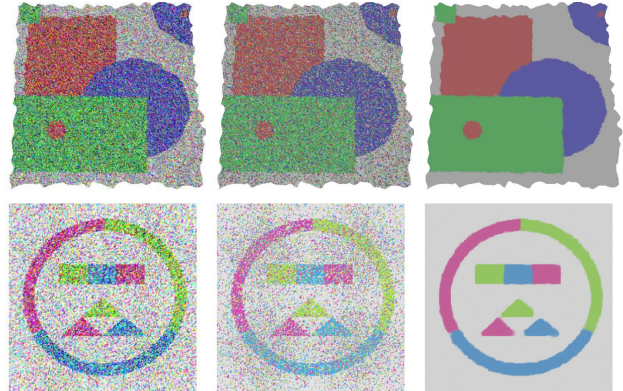


Figure 5: Segmentation results on meshes in the multi-region case on synthesized examples inspired from [24]. (a) Input textured mesh (same shape as previous Butterfly and Horses data). (b) Clustering using K-Means algorithm. (c) Recovered mean values of each region obtained by our approach.

In Figure 5, the experiment shows noisy texture on meshes, the segmentation result using K-Means, and the result of our TV-based algorithm on meshes. Because the K-Means algorithm does not take into account the spatial coherence of points, the result is noisy. On the other hand, the TV regularization allows coherence in the scene and the segmentation is close to the expected solution. In addition to be robust to initialization, our approach is robust to noise.

We then tested our multi-region segmentation approach on various data from real-world images [25], see Figure 6. Let us emphasize here that the initialization was random and the number of regions was initially given.

As an example, we applied our approach to segment mean curvature on a mesh using three different regions. Figure 7 shows that we are able to segment some concave and convex parts of the mesh.

Finally, in Figure 8, we show the examples of a 3D mesh obtained by 3D reconstruction algorithms, as the one in [33, 36]. The last row shows the obtained color-based labeling (into three regions). Even though the texture is far from being binary, the segmentation is the expected one. For instance in the result, we nicely recover the skin, the pant and the shirt. Here again, initialization was random.

For comparison of the convex *image* multi-region segmentation model (3) with other methods, we refer to [24] which shows quantitative and qualitative comparisons with belief propagation, sequential belief propagation, graph cuts



Figure 8: Segmentation result on a colored mesh obtained by multiview stereo algorithm. Front view (top row) and back view (bottom row). Original input colored mesh and the associated 3D shape (left). Result of the segmentation into three regions obtained by our algorithm (middle). Recovered mean values displayed for each region (right).



Figure 6: Segmentation results on meshes in the multi-region case. Top row: Horse data set and its segmentation for three regions. Bottom row: Four regions labeling of the Butterfly data. (a) Input textured mesh; (b) Recovered mean values of each region obtained by our approach; (c) One of the segmented regions.

with alpha-expansion, graph cuts with alpha-beta swap and sequential tree reweighted belief propagation methods. The experiments show that the generated labeling is comparable to state-of-the-art discrete optimization methods.

5. Conclusions

In this paper we propose a variational method for segmenting *data* on *manifolds* into regions of constant properties. The *convex* formulation makes the proposed model

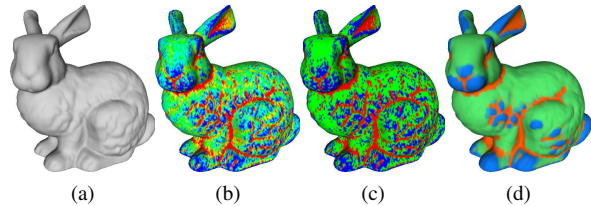


Figure 7: Segmentation results on mesh curvature in 3 regions. (a) Input mesh. (b) Mean Curvature visualization. (c) Simple thresholding of the mean curvature. (d) Segmentation result of the mean curvature into three regions with our approach.

robust to initialization. Moreover, the total variation regularizer makes the method *robust to noise*. We show how to implement the method, in particular how to compute the gradient of the total variation term, when the surface has a discrete representation as triangular meshes. We have demonstrated the efficiency of our method by testing it on various synthetic and realistic data from computer vision applications.

Acknowledgements

Amaël Delaunoy and Emmanuel Prados were supported by the Flamenco project ANR-06-MDCA-007, and Ketut Fundana and Anders Heyden were funded by the VISION-TRAIN RTN CT-2004-005439 Marie Curie Action within the EC's Sixth Framework Programme. We thank Andrei Zaharescu and Adrian Hilton for sharing the dancer data with the Perception group.

References

- [1] E. Aganj, J.-P. Pons, F. Sgonne, and R. Keriven. Spatio-temporal shape from silhouette using four-dimensional delaunay meshing. In *IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, Oct 2007.
- [2] B. Appleton and H. Talbot. Globally minimal surfaces by continuous maximal flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1), January 2006.
- [3] M. Attene, S. Katz, M. Mortara, G. Patanè, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. 2006.
- [4] M. Bertalmio, G. Sapiro, L.-T. Cheng, and S. Osher. Variational problems and pdes on implicit surfaces. *Variational and Level Set Methods in Computer Vision*, 2001.
- [5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.
- [6] Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images. *IEEE International Conference on Computer Vision*, 2001.
- [7] X. Bresson, S. Esedoglu, P. Vanderghynst, J. P. Thiran, and S. J. Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision*, 28(2):151–167, June 2007.
- [8] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, (1), 1997.
- [9] A. Chambolle, D. Cremers, and T. Pock. A convex approach for computing minimal partitions. Technical Report R.I.649, CMAP CNRS, November 2008.
- [10] T. Chan and L. Vese. Active contour without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.
- [11] T. F. Chan, S. Esedoglu, and M. Nikolova. Algorithms for finding global minimizers of image segmentation and denoising models. *Journal of Applied Mathematics*, 2006.
- [12] A. Delaunoy, E. Prados, P. Gargallo, J.-P. Pons, and P. Sturm. Minimizing the multi-view stereo reprojection error for triangular surface meshes. In *British Machine and Vision Conference, Leeds, UK*, 2008.
- [13] A. Demlow and G. Dziuk. An adaptive finite element method for the laplace-beltrami operator on implicitly defined surfaces. *SIAM Journal of Numerical Analysis*, (1), 2007.
- [14] M. P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, February 1976.
- [15] G. Dziuk. Finite elements for the beltrami operator on arbitrary surfaces. In S. B. . Heidelberg, editor, *Partial Differential Equations and Calculus of Variations*, pages 142–155. Lecture Notes in Mathematics, 2006.
- [16] I. Eckstein, J.-P. Pons, Y. Tong, C.-C. J. Kuo, and M. Desbrun. Generalized surface flows for mesh processing. In *Eurographics Symposium on Geometry Processing*, 2007.
- [17] H. Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25, 2003.
- [18] H. Jin, D. Cremers, D. Wang, E. Prados, A. Yezzi, and S. Soatto. 3-d reconstruction of shaded objects from multiple images under unknown illumination. *International Journal of Computer Vision*, 76(3), March 2008.
- [19] H. Jin, A. Yezzi, Y.-H. Tsai, L.-T. Cheng, and S. Soatto. Estimation of 3d surface shape and smooth radiance from 2d images: A level set approach. *J. Sci. Comput.*, (1-3), 2003.
- [20] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.
- [21] P. Kohli and P. H. S. Torr. Dynamic graph cuts for efficient inference in markov random fields. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Dec. 2007.
- [22] Kolmogorov and Zabih. What energy functions can be minimized via graph cuts. *IEEE TPAMI: IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26, 2004.
- [23] M. Krueger, P. Delmas, and G. Gimel'Farb. Active contour based segmentation of 3d surfaces. In *European Conference on Computer Vision*, 2008.
- [24] J. Lellmann, J. Kappes, J. Yuan, F. Becker, and C. Schnoerr. Convex multi-class image labeling by simplex-constrained total variation. Technical report, University of Heidelberg, oct 2008.
- [25] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *International Conference on Computer Vision*, volume 2, pages 416–423, July 2001.
- [26] C. Michelot. A finite algorithm for finding the projection of a point onto the canonical simplex of \mathbb{R}^n . *Journal of Optimization Theory and Applications*, 50(1):195–200, 1986.
- [27] D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and variational problems. *Comm. on Pure and Applied Math.*, XLII(5):577–685, 1988.
- [28] D. L. Page, A. F. Koschan, and M. A. Abidi. Perception-based 3d triangle mesh segmentation using fast marching watersheds. *Computer Vision and Pattern Recognition*, 2003.
- [29] N. Paragios and R. Deriche. Geodesic active regions and level set methods for motion estimation and tracking. *Computer Vision and Image Understanding*, 97:259–282, 2005.
- [30] G. Peyre and L. Cohen. Surface segmentation using geodesic centroidal tessellation. *International Symposium on 3D Data Processing, Visualization and Transmission*, Sept. 2004.
- [31] T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers. A convex formulation of continuous multi-label problems. In *European Conference on Computer Vision*, pages III: 792–805, 2008.
- [32] J.-P. Pons and J.-D. Boissonnat. Delaunay deformable models: Topology-adaptive meshes based on the restricted delaunay triangulation. In *IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, USA, Jun 2007.
- [33] J. Starck and A. Hilton. Surface capture for performance based animation. *IEEE Computer Graphics and Applications*, 2007.
- [34] A. Yezzi and S. Soatto. Stereoscopic segmentation. *IEEE International Conference on Computer Vision*, 2001.
- [35] C. Zach, D. Gallup, J.-M. Frahm, and M. Niethammer. Fast global labeling for real-time stereo using multiple plane sweeps. In *Proceedings of Vision, Modeling, and Visualization*, 2008.
- [36] A. Zaharescu, E. Boyer, and R. Horaud. Transformesh: a topology-adaptive mesh-based approach to surface evolution. In *Proceedings Asian Conference on Computer Vision*, Tokyo, Japan, November 2007.