Mechanical Mesh Segmentation and Global 3D Shape Extraction

Technical Report LIRIS
LIRIS Laboratory — UMR CNRS 5205

Vincent Vidal[1,2] Christian Wolf[1,3] Florent Dupont[1,2]
[1]Université de Lyon, CNRS
[2]Université Lyon 1, LIRIS, UMR5205, Villeurbanne F-69622, France
[3]INSA-Lyon, LIRIS, UMR CNRS 5205, F-69621, France

October 20, 2014

Abstract

This paper presents a method for segmenting noisy 2-manifold meshes based on a decomposition into local shape primitives maximizing global coherence. This technique works by partitioning the input mesh into regions which can be approximated by a simple geometrical primitive such as a plane, a sphere or a cylinder. The proposed approach is entirely error-driven, convergence-proven, and does not need to specify a number of segments.

The partitioning is guided by robust shape extractions based on RANSAC sampling and by a global graphical model which regularizes the segmented regions. The final decomposition is based on the minimum of the energy associated with this graphical model.

Obtained segmentations on noisy mechanical meshes outperform other approaches in terms of region contour correctness and consistency with mechanical object decomposition. Applications of this work are reverse engineering, mesh structure analysis, mesh feature enhancement, noise removal, mesh compression, piecewise approximation of mesh geometry, and remeshing.

Keywords

2-manifold; Triangular Mesh Segmentation; Mesh Decomposition; Line of Feature Edges; Graphical Model;

1 Introduction

In this paper, the NP-hard problem of geometric segmentation [1] is addressed, which aims at dividing an input mesh into polygon clusters approximated by a simple shape primitive [7]. Existing approaches typically use planar primitives and higher-degree primitives such as spheres, cylinders, cones and tori to segment meshes, by capturing these local shape primitives or pieces of primitives on the mesh surface. Geometric segmentation is particularly suited for mechanical objects which are more likely to exhibit consistent geometric area than natural objects.

This paper presents a method for the robust geometric segmentation in the presence of noise on the vertex positions, which is appropriate for the reverse engineering, analysis, processing and approximation of noisy mechanical meshes. Noisy mechanical meshes come usually from surface reconstruction algorithms, either using 3D points obtained by a scanner device or using multi-view imagery. The noise on the vertex position has therefore several origins,

among which are sensors internal noise of the capturing device, environment conditions and surface reconstruction methods.

1.1 Related Work

Segmentation of mechanical objects has been widely studied [2]. There are several approaches to address the geometric segmentation, which consist in several tactics to cluster mesh polygons into semantically consistent regions, among which are hierarchical face clustering [8, 3], variational face clustering [5, 19, 20], variational face clustering and model selection [13], RANSAC extraction [17, 14], curvature guided model extraction [7], and spectral clustering with boundary smoothing [21].

In most of approaches, the two key ingredients are the partitioning (labeling) of polygons and the re-estimation (fitting) of shape primitives onto their region. The partitioning step tries to find clusters of polygons with the desired properties such as the total number of regions, the size or the perimeter length of each region (boundary smoothing). The (local) fitting step estimates the parameters of the best shape primitives mostly using least-squares (regression). This fitting can be done over triangles [7, 5, 19, 20, 13] and point clouds [17, 14]. For meshes, the square of the Euclidean distance is often integrated over polygons leading to the $\mathcal{L}^2$ metric [5], which compensates for irregular sampling.

In case of the presence of outliers and noise on the vertex positions, a robust fitting is done over inlier points [17] or inlier triangles [13]. Li et al.'s [14] work suggests that parameter estimation of geometrical primitives should be subject to global constraints to improve the robustness of the fitting. For instance, some plane primitives almost parallel or orthogonal should be forced to be parallel or orthogonal. However, global parameter estimation is much more computationally expensive than local estimation, and such global priors should be handled with care, since only an expert user may fix such constraints.

Model selection, which consists in finding the best description of the data, is a key step in recent segmentation frameworks. Li et al. [13] have improved the global consistency of the final decomposition by optimizing the number and type of geometrical primitives according to the MDL (*Minimum Description Length*) criterion. However, their optimization is not global, which may result in poor local minima. The global energy approach of Isack and Boykov [11], for robust extraction of lines in 2D point clouds, optimizes the number of consistent lines extracted in a noisy point clouds, and can deal with higher level of noise than approach based on greedy selection of primitives with the most of inliers such as RANSAC.

1.2 Contributions

This paper presents a new method for the partitioning of 2-manifold triangular meshes, which is robust to the presence of outliers and noise on the vertex positions. To our knowledge, our framework introduces two novelties :

- the joint segmentation of mesh triangles and mesh edges in order to favor lines of feature edges pass along region boundaries ;
- and the underlying global (graphical) model which sets the label dependencies and the regularization power of each soft constraint.

Our framework optimizes a globally coherent model including the following soft constraints:

- *label coherence of neighboring triangles*: if an edge is not a feature, adjacent triangles should be associated with the same label (spatial regularization prior) ; on the contrary, in case of a feature edge, adjacent triangle labels should be different ;
- *label cost*: each type of shape primitives has its own cost (label cost prior), thus allowing the global optimization of the type and number of shape primitives.

1.3 Segmentation algorithm overview

The problem of geometric segmentation is clearly continuous. Indeed, each type of shape primitive is described by a set of real-valued parameters. We therefore resort to a discretization of the problem, i.e. manage a set of candidate primitives, which involves the use of three strategies in an iterative process:

- *initial discretization is based on RANSAC*: according to mesh triangle barycenters and normals, a set of best primitive candidates is extracted ;
- *globally coherent partitioning* using current set of shape primitives ;
- *discretization is refined according to the current partition*: if the refitting of a shape primitive onto its inlier triangles leads to a noise variance greater than a predefined maximum variance, new shape primitives are extracted using RANSAC, otherwise the refitted primitive is added to the primitive candidates.

Using our discretization to the geometric segmentation problem, our global model (see section 2) permits to automatically select an excellent subset of shape primitives. Our framework is general, as it encompasses MDL-like criteria: the type and number of shape primitives are globally optimized according to user requirements. In addition, shape primitives are not restricted to approximate connected set of triangles, which permits to decompose a mesh with fewer shape primitives, in particular for mechanical objects where some shape primitives are an extension of others. For instance, the gear model in figure 2, the noisy fandisk model in figure 3, and the mechanical model in figure 8 have a cylindrical proxy associated with a 2-connected region.

Our mesh segmentation model is presented in the next section and the following section deals with the energy function related to our graphical model. Section 4 addresses the minimization of the energy function and the management of shape primitive candidates. Afterwards, experimental results on noisy mechanical meshes and scanned objects are discussed.

2 Model for Mesh Segmentation and Shape Extraction

In our work, mesh segmentation is coupled with extraction of lines of feature edges and is formulated as an energy minimization problem, where triangles and edges are labeled. Each triangle label corresponds to a shape primitive approximating its region and each binary edge label indicates (or not) the presence of a feature edge along a feature line. In the following, we introduce relevant definitions and notations used throughout the paper, as well as the model variables and its goals.

2.1 Definitions and Notations

A *feature* edge is either a boundary edge or an edge along which a sharp discontinuity of the normal field occurs. A *region* is a set of triangles associated with the same label. In contrast to frequent definitions, non-connected triangles are allowed to be clustered together. As in previous work [5, 19, 13], a shape primitive approximating a region is called *shape proxy* or *proxy* for short.

The following notations are used:

- a capital letter X represents a family of random variables, while a lower case letter x represents a realization of X ;
- a subscripted capital letter X_i represents a random variable associated with the node i of a graphical model, while a subscripted lower case letter x_i represents a realization of X_i.

2.2 Model Variables

For the joint segmentation of triangles and edges, we propose an undirected graphical model, whose nodes are divided into two categories:

- $\mathcal{S}^{(t)}$ represents the nodes associated with mesh triangles ; the neighboring relationship is induced by the mesh: two nodes are neighbors **iff** the two corresponding mesh triangles share a common mesh edge ;
- $\mathcal{S}^{(x)}$ are nodes linked to mesh edges ; the neighboring relationship is induced by the mesh: two nodes are neighbors **iff** the two corresponding mesh edges share a common mesh vertex.

The corresponding edges of this graphical model are:

- $\mathcal{E}^{(t)}$ denotes the set of edges between two neighboring nodes in $\mathcal{S}^{(t)}$;
- $\mathcal{E}^{(x)}$ accounts for the set of edges between two neighboring nodes in $\mathcal{S}^{(x)}$.

The set of edges $\notin \mathcal{E}^{(t)} \bigcup \mathcal{E}^{(x)}$, which are between triangle and adjacent edge sites, is not associated with a specific notation. Note that edges are undirected and thus (i, j) and (j, i) represent the same edge. Each node of our graphical model is associated with a discrete random variable. Let T and X be, respectively, a family of random variables over sites $\mathcal{S}^{(t)}$ and $\mathcal{S}^{(x)}$. Let G and Y be, respectively, a family of observed variables over sites $\mathcal{S}^{(t)}$ and $\mathcal{S}^{(x)}$. Observed variables integrate sensed geometrical features with respect to our global model. g includes the geometry of mesh triangles needed for computing the approximation error between the geometric triangle g_i and its proxy $\mathcal{P}_{t_i}$, while y is made of all geometrical features used in the model of line of feature edges. $\mathcal{P}_{t_i}$ (resp. $\mathcal{P}$) is the proxy of the triangle i (resp. the set of all proxies).

2.3 Model Objectives

The goals of our graphical model are threefold:

- to label triangles with a label corresponding either to a shape proxy index $\{0,\ldots,|\mathcal{P}|\text{-}1\}$ or to -1 ; in the latter case, the kind of proxy is $\varnothing$; here -1 label means either outliers or cannot be well approximated by current proxy candidates ;
- to label mesh edges by $x_i \in \{$*non-feature*=0, *feature*=1$\}$;
- to make triangle and edge labeling consistent with each other, by modeling the joint distribution of triangle and edge labels.

2.4 Segmentation Pipeline

The whole segmentation process is illustrated in figure 1. A set of shape proxies is initially obtained by a RANSAC discretization of the continuous space of shape primitive parameters. Then, a subset of these shape proxies is assigned to triangle variables by a joint labeling of triangles and edges. More precisely, the labeling process will set triangle variables to a shape proxy index and will set edge variables either to *non-feature* or *feature*. Afterwards, the current segmentation is used to update the current discretization by adding new shape proxies and removing unused ones in the set of candidate primitives. The algorithm terminates when the set of candidate proxies and the current segmentation are stable, which occurs at a (good) local minimum of the energy minimized. During the segmentation process, all proxy selection steps are based on the global minimization of the objective function (1), presented in the next section, and thus each triangle label change is related to an energy decrease.

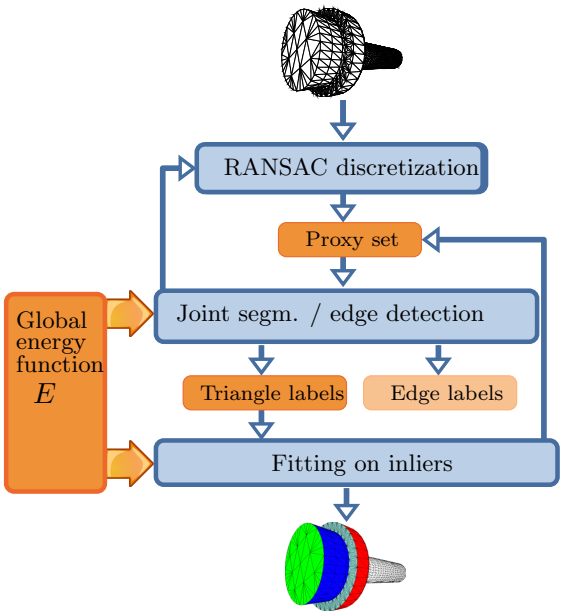


Figure 1: Pipeline of our method. A first RANSAC model extraction initializes the set of geometrical primitives. A global selection is then completed minimizing the objective function (1). Based on the current mesh segmentation, a new set of candidates is added to the proxy set. Eventually, when the set of primitives and the mesh segmentation are stable, a final primitive fitting is done to decrease further the energy E. The possibility to remove a shape primitive from the set of candidate primitives is not drawn here for clarity purpose.

3 Objective Function

The objective function (1) assigns an energy to each possible solution of the global labeling problem and evaluates to lower energy values for more favorable solutions. It is a weighted scalar sum of scale-invariant energy potentials, where five types of energy potentials appear:

- $E_{feat}(x_i; y)$ is the local evidence of edge $i \in \mathcal{S}^{(x)}$ being a feature edge or not and is defined in equation (3) ; it is based on machine learning and takes into account the prediction from a statistically learnt model to be less sensitive to the presence of noise on the vertex positions ;

- $E_{dir}(x_i, x_j; \theta, \phi)$ is a spatial regularization model on couples of neighboring edge sites $(i, j) \in \mathcal{E}^{(x)}$, which favors almost aligned couples of feature edges which have similar dihedral angle and is defined in equations (4) and (5) ;

- $E_{approx}(t_i; g_i; \mathcal{P}_{t_i})$ is called *residual error* and measures the approximation error between the geometric triangle g_i from site $i \in \mathcal{S}^{(t)}$ and its shape proxy $\mathcal{P}_{t_i}$ and is defined in equation (2) ;

- $E_{reg}(t_i, t_j, x_{\psi(i,j)})$ is a spatial regularization prior on triplet $(t_i, t_j, x_{\psi(i,j)})$ of two adjacent triangles $i, j \in \mathcal{S}^{(t)}$ and their shared edge $\psi(i, j) \in \mathcal{S}^{(x)}$, and is defined in equation (6) ;

- $\beta_k |\mathcal{L}^{(L_k)}|$ penalizes the proxy use of type $k \in \{\varnothing, plane, sphere, cylinder\}$ by the positive cost β_k ; $\mathcal{L}^{(L_k)}$ is the set of indices of shape proxy of type k used in the current configuration $L = \bigcup_k L_k$; $|.|$ is the number of elements in the set (cardinality) ; $\beta_k |\mathcal{L}^{(L_k)}|$ serves as an automatic model selection method.

The energy function is given as

$$\begin{aligned} E(t;x;g;y;\mathcal{P}) \quad &= \\ \kappa \quad &\sum_{i\in\mathcal{S}^{(x)}} \quad E_{feat}(x_i;y) \\ +\lambda \quad &\sum_{(i,j)\in\mathcal{E}^{(x)}} \quad E_{dir}(x_i,x_j;y) \\ +\rho \quad &\sum_{i\in\mathcal{S}^{(t)}} \quad E_{approx}(t_i;g_i;\mathcal{P}_{t_i}) \\ + \quad &\sum_{(i,j)\in\mathcal{E}^{(t)}} \quad E_{reg}(t_i,t_j,x_{\psi(i,j)}) \\ + \quad &\sum_k \quad \beta_k|\mathcal{L}^{(L_k)}| \end{aligned} \tag{1}$$

where κ is a constant adjusting the strength of edge attachment terms of data compared with those of the triangles. λ is a positive scalar used to control the regularization strength of the edge spatial prior. ρ is a positive weight, which permits to monitor the strength of the approximation error with respect to regularization priors. The terms $E_{reg}(t_i,t_j,x_{\psi(i,j)})$ have no visible weight in the equation (1) because their weights are internally parameterized. β_k costs offer a fine control onto the number of shape proxies of each type, but also permit a more compact description of data through the merging of proxies close enough.

3.1 Triangle Residual Error

The triangle residual error $E_{approx}(t_i;g_i;\mathcal{P}_{t_i})$ depends on the label t_i of a site $i \in \mathcal{S}^{(t)}$, on the geometrical triangle g_i and on the shape proxy $\mathcal{P}_{t_i}$ among all proxy candidates $\mathcal{P}$:

$$E_{approx}(t_i;g_i;\mathcal{P}_{t_i}) = \begin{cases} \mathcal{L}^2(g_i,\mathcal{P}_{t_i}) & \text{if } \mathcal{P}_{t_i} \neq \varnothing \\ \gamma \times |g_i| & \text{else} \end{cases} \tag{2}$$

where $\mathcal{L}^2(g_i,\mathcal{P}_{t_i}) = \int_{p\in g_i} \|p - \prod_i(p)\|^2 dp$ is a metric which measures the error between the geometrical triangle g_i and its shape proxy $\mathcal{P}_{t_i}$, and $\prod_i(p)$ denotes the orthogonal projection of point p onto proxy $\mathcal{P}_{t_i}$. As in VSA [5], the normalization by the region area and the square root have been removed, because they are not useful for optimization purpose. An exact formula for $\mathcal{L}^2(g_i,\mathcal{P}_{t_i})$ is given in [5] in case of planar proxies. This formula is also used in case of spherical or cylindrical proxies as a good approximation [19]. $|g_i|$ means the area of triangle g_i. $\gamma \times |g_i|$ is the cost of a non-associated triangle, it depends on the triangle area as does the $\mathcal{L}^2$ metric. γ is a positive constant which controls the distance of a triangle to its approximating proxy: if γ is large enough, all triangles will be associated to a proxy candidate. On the contrary, if γ is close to zero, a large amount of triangles may remain non-associated along with the segmentation process. To achieve results invariant to scale, the input mesh is normalized such that the largest dimension of its axis aligned bounding box is 1. At the end of the segmentation, the mesh is denormalized.

3.2 Feature Edge Model

The feature edge model included in our global model is a CRF for robust extraction of lines composed of feature edges, in the lines of the one proposed by [18]. $E_{feat}(x_i;y)$ is a data-attached term which gives lower energy for a labeling x_i which is consistent with the response of a (learnt) prediction model according to observed geometrical features y.

$$E_{feat}(x_i;y) = exp((-1)^{x_i} d(y)) \tag{3}$$

$E_{feat}(x_i;y)$ directly incorporates classification and confidence of a SVM (Support Vector Machine) — which depends on the obtained prediction model — through the signed distance $d(y)$

of the mapped feature vector associated with edge i to the separating hyperplane (in feature space). It is positive for predicted feature edges and negative otherwise. The feature vector is made of 43 geometrical features such as the edge dihedral angle or the edge mean curvature, and the SVM, a machine learning algorithm, learns how to characterize a feature edge. We employed the same features as in [18].

$E_{dir}(x_i, x_j; y)$ is a directional regularization term on couples of neighboring edge sites, which gives lower energy to a labeling (x_i, x_j) when $x_i = x_j$, while favoring the case of two feature edges only when they are rather aligned. The edge coherence term is given by

$$E_{dir}(x_i, x_j; y = [\theta_i, \theta_j, \phi_{\psi'(i,j)}])) = \begin{cases} 0 & \text{if } x_i \neq x_j \\ -\tau & \text{if } x_i = x_j = 0 \\ -\Gamma(\theta_i, \theta_j, \phi_{\psi'(i,j)}) & \text{else} \end{cases} \tag{4}$$

where τ is a positive constant which controls the homogeneity of normal edges. $\theta_i \in y$ is the dihedral angle of edge i. $\phi_{\psi'(i,j)} \in y$ is the turning angle between edges i and j. Γ (cf. equation 5) is a positive functional used in the 2 feature edges case, which reaches larger values for geometrically aligned (within a tolerance) neighboring edges if they have similar dihedral angles.

$$\Gamma(\theta_i, \theta_j, \phi_{\psi'(i,j)}) = exp\left\{ -\Delta \left(\underbrace{\frac{|cos(\theta_i) - cos(\theta_j)|}{\sigma}}_{similarity} + \underbrace{1 - cos(\phi_{\psi'(i,j)})}_{alignment} \right) \right\} \tag{5}$$

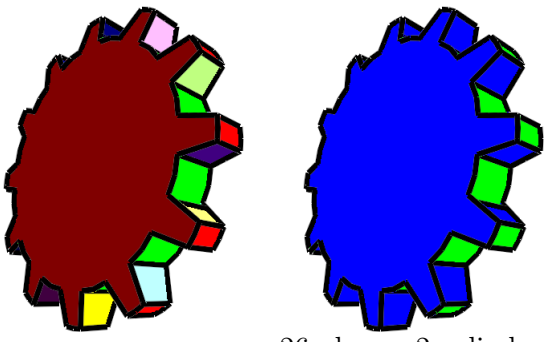
In equation (5), Δ is a positive constant which monitors the neighboring edge alignment tolerance. The greater Δ is, the more two neighboring edges must be aligned. σ handles the tolerated variance of $\Delta|cos(\theta_i) - cos(\theta_j)|$ along feature lines. The similarity term avoids detecting adjacent feature edges with a very different dihedral angle. The alignment term predominates over similarity term, for favoring non-jagged feature line extraction.

3.3 Consistency between Triangle and Edge Labels

The goal of the consistency term is to favor equal (resp. different) triangle labels between two neighboring triangles when their shared edge is not (resp. is) a feature. The consistency term is defined as follows:

$$E_{reg}(t_i, t_j, x_{\psi(i,j)}) = \begin{cases} \mu & \text{if } t_i \neq t_j \text{ and } x_{\psi(i,j)} = 0 \\ \nu & \text{if } t_i \neq t_j \text{ and } x_{\psi(i,j)} = 1 \\ \zeta & \text{if } t_i = t_j \text{ and } x_{\psi(i,j)} = 0 \\ \eta & \text{if } t_i = t_j \text{ and } x_{\psi(i,j)} = 1 \end{cases} \tag{6}$$

where μ, ν, ζ, and η are positive constants which penalize, respectively, the label difference of two triangles sharing a normal edge, the label difference of two triangles sharing a feature edge, the label equality of two triangles sharing a normal edge, and the label equality of two triangles sharing a feature edge. Note that when $\mu = \nu \geq 0$ and $\zeta = \eta = 0$, the consistency term corresponds to the Potts/Ising model [9], independent of the type of an edge, which is used by the PEaRL algorithm [11]. Therefore, our proposed consistency term offers richer interactions. However, because we want to favor equality (resp. inequality) of the labels t_i and t_j of two neighboring triangles only when the shared edge is not (resp. is) a feature, we restrict the parameter setting to $\mu = \nu > 0$, $\zeta = 0$ and $\eta \geq \mu, \nu, \zeta$. Intuitively, this setting means that the consistency term:



gear 26 planes, 2 cylinders

Figure 2: Results of our method on a gear (with feature edges): (left) one color per shape primitive index and (right) one color per shape primitive type (blue = plane, green = cylinder). Our method can segment non-noisy mechanical models by increasing the cost related to the approximation error. Two cylinder proxy associated with non-connected regions are successfully found by our framework.

- penalizes the difference of neighboring triangle labels,
- and penalizes even more the equality of neighboring triangle labels when their shared edge is a feature.

4 Minimization and Proxy Management

In general, in discrete optimization, better guarantees can be obtained in the binary case (two possible values for each variable) than in the multi-label case which is often solved using successive binary decisions between keeping the current label or taking the new one.

In the binary case, where only two shape proxies $\mathcal{P}_0$ and $\mathcal{P}_1$ are available, when $E_{reg}(t_i, t_j, x_{\psi(i,j)})$ is submodular, the objective function (1) can be globally minimized using the *graph cuts* framework, especially Kolmogorov and Zabih's st-graph construction method [12]. The submodularity of E_{reg} depends on the parameters of the functional and means that

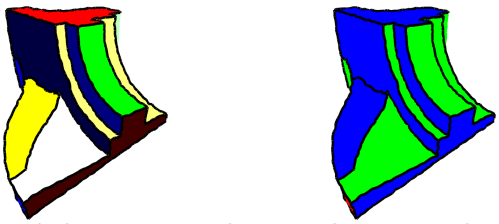
$$E_{reg}(0,0,0) + E_{reg}(1,1,0) \leq E_{reg}(0,1,0) + E_{reg}(1,0,0) \tag{7}$$

and

$$E_{reg}(0,0,1) + E_{reg}(1,1,1) \leq E_{reg}(0,1,1) + E_{reg}(1,0,1). \tag{8}$$

Submodularity (cf. [12]) is an important concept in discrete optimization. Its meaning is the equivalent of the concept of convexity, restricted to discrete sets. Equation (7) requires $\zeta \leq \mu$ which is always true. Equation (8) requires $\eta \leq \nu$, which is generally false, since we would like to penalize more the case of triangle label equality with a feature edge. Consequently our energy function is not submodular. Fortunately, there exist minimization techniques able to deal with non-submodular terms such as BP (Belief Propagation) [15] or QPBO (Quadratic Pseudo-Boolean Optimization) [16]. While both methods give approximated results on non-submodular functions, the convergence of BP is not guaranteed in the presence of cycles in the graphical model, which leads us to choose QPBO whose convergence is guaranteed. In addition, QPBO gives optimal results on submodular functions.

In the multi-label case (at least 3 proxies), successive binary decisions between keeping the current label or choosing the new label (one loop over all possible labels for one iteration) are taken using QPBO. If the objective function (1) is submodular, a strong local minimum is found by successive QPBO, that is to say a minimum within a factor of the global optimum



noisy fandisk 13K triangles

8 planes, 2 spheres, 6 cylinders

noisy fandisk 52K triangles

8 planes, 2 spheres, 6 cylinders

Figure 3: Results of our method (with feature edges): (left) one color per shape primitive index and (right) one color per shape primitive type (blue = plane, red = sphere, green = cylinder). Segmented models show that our method is robust to the sampling density once there are enough proxy inliers. Our framework permits to extract a cylinder over 2 non-connected regions while VSA-like approaches will use two cylinders instead.

(see [4], section 6). QPBO guarantees that after each binary decision the global energy does not increase, which proves the convergence of our segmentation algorithm via the minimization approach presented in section 4.1.

A last advantage of QPBO, in both binary and multi-label cases, is that it experimentally finds a good local minimum when the number of non-submodular terms is low.

4.1 Minimization

The objective function (1) is hard to minimize, because it is not submodular for selected parameters. In the proposed method, (1) is iteratively minimized by α-expansion move with label cost [6]: starting from an initial configuration, and for each available triangle label, a binary decision is done for each triangle between keeping the current label or the new one. More precisely, for each binary decision (1) is approximately minimized with QPBO [16].

Noting that in general the number of normal edges is largely dominant on the number of feature edges, and that for normal edges, consistency terms are submodular, we finally opt for a two steps minimization procedure: first of all, the line of feature edges are extracted using the method in [18] ; then, shape primitive extraction and global selection steps are iterated until convergence: the selection step will use our graphical model, while observing edge labels. Because QPBO works well when the number of non-submodular terms is low, an excellent local minimum is generally found. Note that the method for feature line extraction is now independent from the global model and thus another line extraction method may be used.

4.2 Management of Proxy Candidates

The management of proxy candidates begins with the initial proposal of shape proxies, the discretization step: starting from a triangle soup, the largest set of triangles supporting either a planar, a spherical or a cylindrical proxy is found by a RANSAC. Then, a shape primitive fitting is done on these inliers triangles using least-squares and the obtained shape proxy is added to the set of proxy candidates. The discretization continues on the remaining triangles

up to no more shape proxy can be extracted. Once the initial proxy set is found, it will be modified by proxy additions (section 4.2.1) and proxy removals (section 4.2.2).

4.2.1 Refined Discretization according to Mesh Segmentation

After a labeling step, each segmented region associated with a shape proxy is analyzed as follows:

1. current proxy parameters are re-estimated in the least-square sense over all region triangles ;
2. parameters are checked to make sure they are meaningful: if a spherical or cylindrical proxy has a radius greater than a specified maximum radius, the best planar proxy is extracted instead of current proxy, using RANSAC ;
3. the last step is an intra-region variance monitoring of the newly estimated proxy: if the current new proxy is associated with a variance below a maximum variance, this new proxy is added to the set of proxy candidates ; otherwise, the best proxy of each type is extracted by a RANSAC and added to the list of candidate proxies ; in the case of a planar proxy with too high variance, the two best planar proxies are added instead of the best planar proxy.

For regions labeled as $\varnothing$, i.e. non-associated, the best proxy of each type is extracted by a RANSAC and added to the list of proxy candidates. Note that at this point, no decision to use a newly proposed proxy has been fixed. This will be done in the next joint segmentation step that will make the final decision among old and new candidates. One may expect to propose additional proxy candidates based on the merging of neighboring regions. However, assuming that RANSAC extraction has found all proxies associated with intermediate size region, this possibility is not necessary.

4.2.2 Shape Proxy Removals

A high number of proxy candidates may degrade the performances of our segmentation algorithm, because the chosen algorithm for discrete optimization is linearly dependent on the number of triangle label candidates. As proposed by Li at al. [13], a shape proxy candidate not supported by any triangle of the initial mesh during two successive steps of proxy discretization and global selection, should be removed. This heuristic works well in practice and experimentally outperforms similar heuristics using different values for the number of successive whole steps.

However, we found by experimentation that sorting in decreasing order the set of candidates according to their number of support points allows to run only one α-expansion (with label cost) for the selection step (cf. section 4.1).

5 Experimental Results

To demonstrate the efficiency of our method, we applied it to several mechanical models with noise on their vertex positions. More specifically, there are 11 normalized 2-manifold meshes among which 5 are obtained by adding 0.5% Gaussian noise intensity to mesh positions and the 6 remaining models are composed of 4 scanned objects and 2 mechanical objects. The normalization process consists in mesh scaling to make the maximal dimension of a mesh bounding box equal to one, and as mentioned earlier is needed to make our segmentation results scale-invariant. 0.5% thus refers to the maximal bounding box dimension.

The presented results have been obtained on an Intel Core I7-2760QM (2.4 GHz) with 8GB RAM, with a basic RANSAC sampling (not the one in [17]).

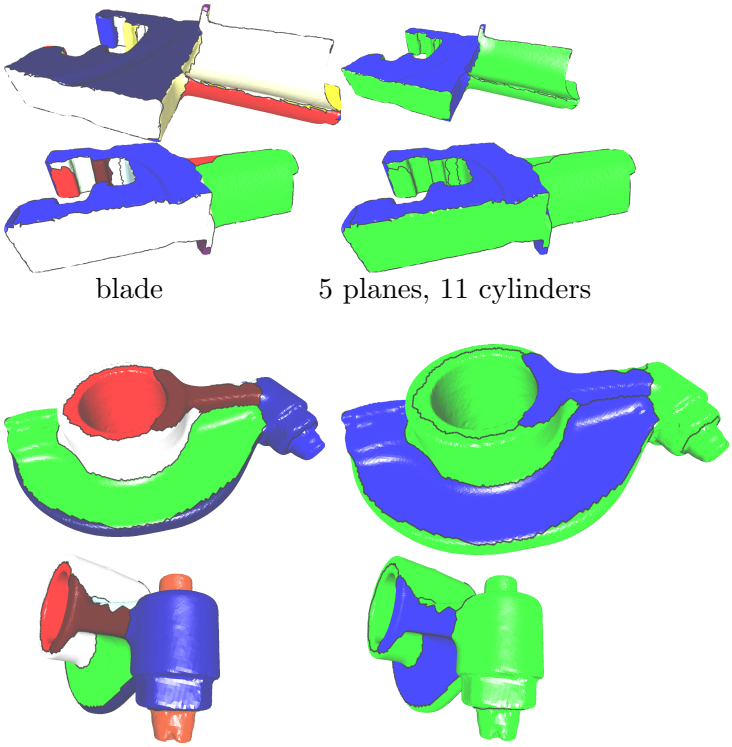


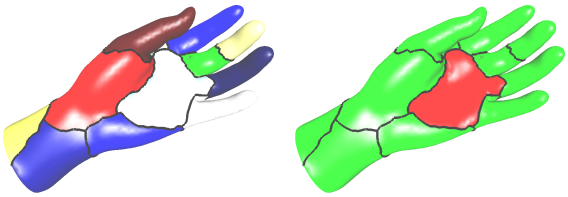
Figure 4: Results of our method: feature lines extracted using dihedral angle have been used for the blade model, while no feature line has been used for the rockerArm model: (left) one color per shape primitive index and (right) one color per shape primitive type (blue = plane, green = cylinder). Feature line information avoid region overflowing.

Models	#t (K)	#e (K)	Time (s)	#p	#s	#c
cylinder	2.36	3.54	3	2	0	1
screw	2.48	3.72	2	4	0	3
joint	6.02	9.02	13	9	0	3
sphere	9.80	14.70	5	0	1	0
fandisk	12.99	19.48	31	8	2	6
blade	25.20	37.80	373	5	0	11
rockerArm	80.35	120.53	988	3	0	8
hand	11.69	17.54	66	0	1	12
bunny	69.67	104.50	377	2	6	2
mech	1.86	2.78	5	6	0	2
gear	5.31	7.97	56	26	0	2

Table 1: Statistics on the segmented models: number of triangles and edges, the running time, and the number of planar, spherical and cylindrical primitives in the final segmentation.

$\kappa = 1$	$\lambda = 0.2$	$\rho = 11150$
$\beta_{plane} = 0.102$	$\beta_{sphere} = 0.110$	$\beta_{cylinder} = 0.106$
$\beta_{\varnothing} = 0.102$	$\gamma = 10^{-2}$	-
$\tau = 0$	$\Delta = 15$	$\sigma = 10$
$\mu = \nu = 6.10^{-3}$	$\zeta = 0$	$\eta = 8.10^{-3}$

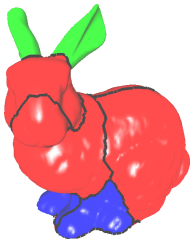
Table 2: The parameters of our model (cf. equations 1, 2, 4, 5, and 6). $\rho = 26650$ (resp. 30000) in figure 2 (resp. figure 4).



hand

0 plane, 1 sphere, 12 cylinders

bunny



2 planes, 6 spheres, 2 cylinders

Figure 5: Results of our method (without feature edge): (left) one color per shape primitive index and (right) one color per shape primitive type (blue = plane, red = sphere, green = cylinder). Our method works well on natural objects: the symmetries of the bunny model are preserved in its decomposition and the fingers of the hand model are associated to either one or two cylinders depending on the finger' bending.

5.1 Discussion

To illustrate the ability of our mesh segmentation method to capture the right decomposition of a mechanical object, we applied it onto simple noisy meshes, whose ideal decomposition is obvious: cylinder, screw, joint and sphere. Table 1 and figure 6 confirm that the obtained decompositions are the right ones. To evaluate the ability to segment more complex mechanical objects, we applied our method on a gear model (see figure 2), on the noisy fandisk models (see figure 3) and on two scanned mechanical objects (see figure 4). We also applied it to two scanned natural objects for which no feature edge has been extracted (see figure 5). Visual results show that our method is able to extract a compact surface description made of shape proxies with a large support. Regions approximated by a proxy may be non-connected. If the end-user needs connected regions, a simple post-processing step which extracts connected components can be used. In addition, region complexity is kept low and contour smoothness is encouraged by our method, because of the spatial regularization prior which favors compact regions with as few as possible jagged contour edges. Note that if crest lines are available for natural objects, their use in our algorithm as feature lines input will make the final segmentation respecting the minima-rule [10], i.e. a crest line will be at the junction of two different regions.

The joint segmentation of mesh triangles and edges permits to attract region boundaries

towards lines of features edges, which improves the global coherence of final results, and also avoids overflowing of one region label across the line of feature edges. In addition, our method is not sensitive to isolated feature edges, that is, region boundaries pass along relevant line of feature edges if there is a line of feature edges close enough.

Our geometric segmentation can cope with noise on the vertex positions, and is robust to irregular (non-uniform) sampling, in particular to remeshing. The extracted mesh decompositions in figure 3 for the fandisk model are indeed very similar for two different sampling rates. The natural model decomposition into simple shape primitives is therefore not sensitive to the density of mesh vertices if a sufficient number of inliers for each mesh region is present (for RANSAC). However, if the region boundaries are affected by a mesh pre-processing, the final segmentation may slightly differ.

5.1.1 Comparisons with previous work

We compare our method to HFC [3] and VSA [5, 19] approaches using our own implementation based on the fitting over mesh triangles (in original HFC paper, the fitting is done over points). We use our own implementation, because the proxy fitting influences the final segmentation ; thus by using the same fitting algorithm for the 3 methods we are able to compare the 3 frameworks without either the fitting method or fitting implementation bias. HFC and VSA need the user to set a number of segments to work, whereas our method does not need it. As can be seen in figure 7, our method gives better results in terms of region complexity and contour smoothness due to the absence of spatial regularization prior in HFC and VSA methods. The worst region boundaries are obtained for HFC, because of its greedy nature which can get stuck into a poor local minimum. Note in the same figure that VSA may suffer from small island configuration, while it does not experimentally appear for our method. In figure 8 (a), we can see that our framework is able to recover shape proxy associated with a non-connected region (the red and white cylinders), while HFC and VSA are not able to do so. Several previous works applied their method to the bunny model [3, 19, 20, 13]. On average, the segmented bunny with our method without feature line has a more compact description with region boundaries less complex. Our method seems also to preserve the symmetries on the bunny model.

5.2 Implementation Details and Settings

Parameters have been experimentally fixed with the values presented in table 2. Penalizing less a proxy type is similar to taking into account *a priori* knowledge about more probable proxy type. Cylindrical proxies are usually more present in mechanical objects than spherical proxies, that is why the cost of a spherical proxy is greater than the cost of a cylindrical proxy. Penalizing more a proxy will also favor the merging of two close proxies of the same type, which is the case with the chosen label costs. By increasing all label costs, an end-user can obtain another level of detail for the mesh segmentation. However, the best way to control different levels of detail is by the weight ρ used to penalize the residual error between triangles and their proxy.

Another relevant *a priori* knowledge is the maximum radius size of kept spherical and cylindrical proxies, which is set to $7 \times MBB$ for both. MBB is the maximal bounding box dimension. Indeed, without such an information, a planar area may be approximated by a cylindrical proxy whose radius is very large.

Let mBB be the minimal dimension of the mesh bounding box. Concerning the RANSAC thresholds needed for the initial primitive extraction, we experimentally set the distance threshold to $4.10^{-2} \times mBB$ and the deviation threshold to 14.07 degrees. In the case of RANSAC extraction based on the current segmentation, we experimentally set the distance threshold to $8.10^{-3} \times mBB$ and the deviation threshold to 11.48 degrees. For the shape proxy

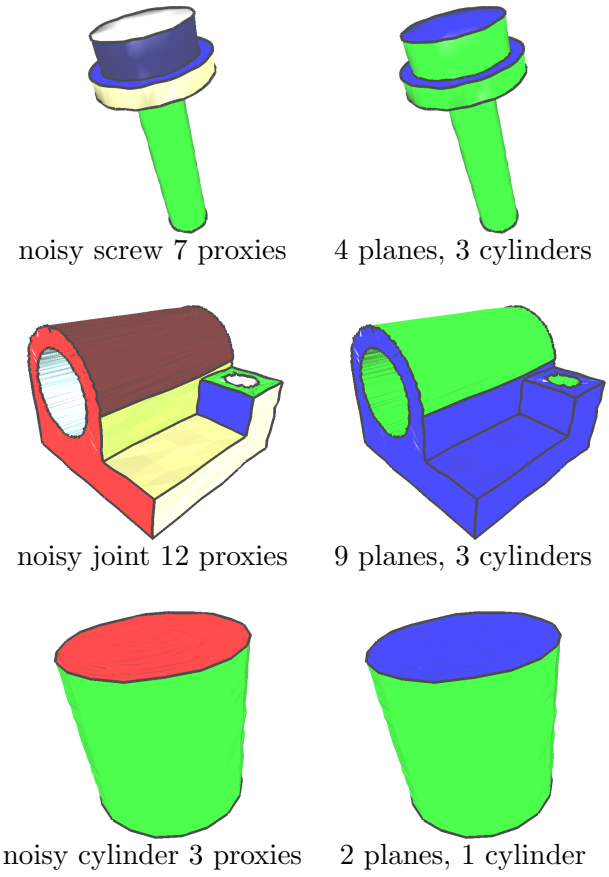
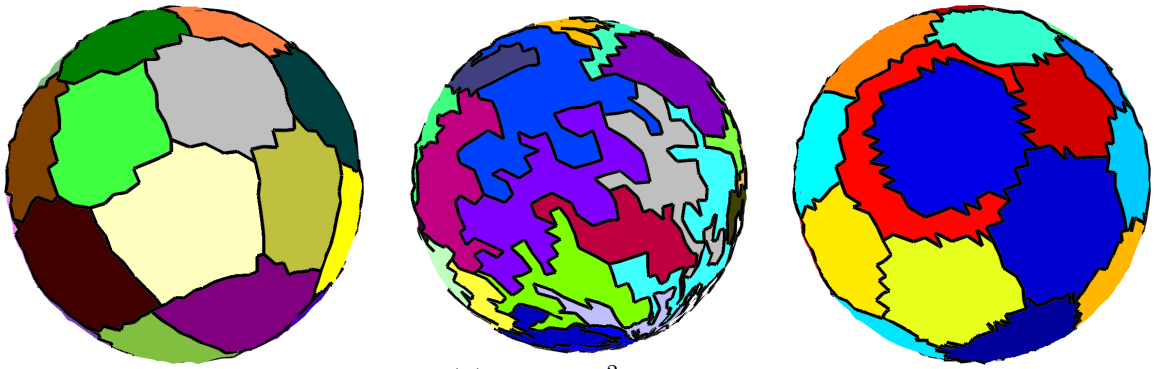
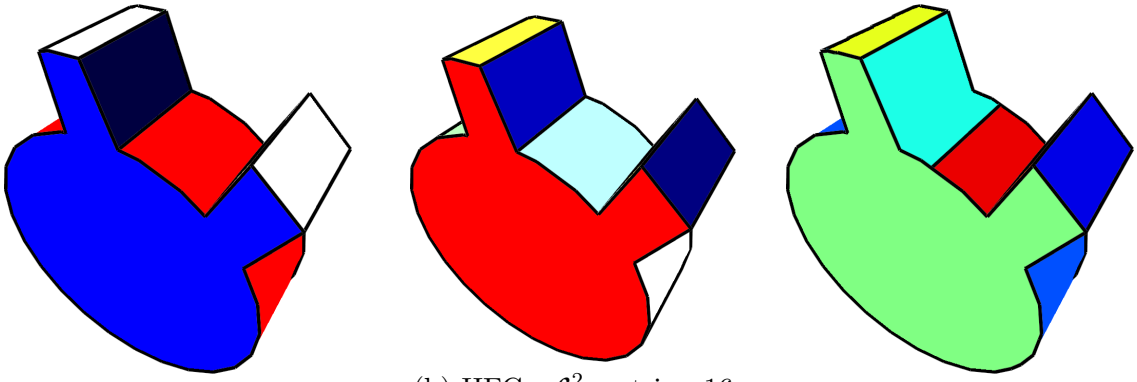


Figure 6: Results of our method on noisy mechanical objects (with feature edges): (left) one color per shape primitive index and (right) one color per shape primitive type (blue = plane, red = sphere, green = cylinder). Our method finds the right decomposition of simple mechanical objects.



(a) Our method - $\mathcal{L}^2$ metric (b) HFC - $\mathcal{L}^2$ metric (c) VSA - $\mathcal{L}^2$ metric

Figure 7: Piecewise-linear approximation of a noisy sphere (0.5% Gaussian noise) using (a) our method, (b) HFC and (c) VSA. We explicitly restrict all methods to only use planar proxies in order to analyze the quality of partitions. Our method automatically finds the best number of planar proxies (40), while for HFC and VSA this number was set as an input.



(a) Our method - $\mathcal{L}^2$ metric - 6 planes and 2 cylinders

(b) HFC - $\mathcal{L}^2$ metric - 16 planes

(c) VSA - $\mathcal{L}^2$ metric - 7 planes and 3 cylinders

Figure 8: Shape approximation of a mechanical model using (a) our method, (b) HFC and (c) VSA. Our method automatically finds the best number of proxies, 8, while for VSA and HFC we experimentally found 10 and 16 as the best solution. Our framework describes the mesh geometry with fewer proxies and permits to extract a cylinder over 2 non-connected regions while VSA-like approaches will use two cylinders instead.

initial estimate, needed for RANSAC, we used those proposed in [17]. For the planar proxy fitting, we use the classical method based on Principal Component Analysis [3]. For the fitting of spherical and cylindrical proxies, starting from the initial estimate, we use conjugate gradient minimization for triangle soups with less than 90 triangles, and otherwise we use the Attene et al.'s fitting method [3], a computationally efficient approach. With a high number of triangles, Attene et al.'s fitting method is sufficiently accurate, while for a low number of points gradient descent is much more accurate.

For the α-expansion move with label cost [6], which is an energy minimization algorithm able to minimize an energy function in the multi-label case, we use our implementation based on the QPBO graph cut technique given in Rother et al. [16].

6 Conclusion and future work

In this paper, we have presented a graphical model for the geometric segmentation of 2-manifold triangular meshes, coupled with the extraction of lines of feature edges. The obtained results compare favorably to previous work: our method is not sensitive to noise on the vertex positions, not sensitive to irregular surface sampling and is able to capture the right decomposition of simple mechanical objects, without specifying the number of regions.

As future work, we will investigate learning of our model parameters using scanned mechanical objects whose decomposition into shape primitives is known. We would like to include a finer control on the authorized noise variance to cope with meshes with non-uniform noise on their vertex positions. We wish to propose a hierarchical version of our graphical model, to favor consistent decomposition at several scales. We would like to improve the graphical model of line of feature edges, to make it more adapted to scanned objects by taking into account extended dihedral angle integrated along geodesic lines at different scales. Finally, we wish to take into account the area of the extracted regions into the RANSAC score function to favor proxies associated with large area first instead of a large number of inlier triangles.

References

[1] PK Agarwal and S Suri. Surface approximation and geometric partitions. In *Proceedings of the fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 24–33, 1994.

[2] Alexander Agathos, Ioannis Pratikakis, Stavros Perantonis, Nikolaos Sapidis, and Philip Azariadis. 3D mesh segmentation methodologies for CAD applications. *Computer-Aided Design and Applications*, 4(6):827–841, 2007.

[3] Marco Attene, Bianca Falcidieno, and Michela Spagnuolo. Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, 22(3):181–193, February 2006.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.

[5] David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. Variational shape approximation. *Proceedings of the 31st Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 23(3):905–914, 2004.

[6] Andrew Delong, Anton Osokin, Hossam N. Isack, and Yuri Boykov. Fast approximate energy minimization with label costs. In *Computer Vision and Pattern Recognition (CVPR)*, pages 2173–2180. IEEE, 2010.

[7] Julie Digne, J.M. Morel, C. Mehdi-Souzani, and Claire Lartigue. Mesh Segmentation and Model Extraction. *Curves and Surfaces*, 6920:236–252, 2012.

[8] Michael Garland, Andrew Willmott, and P.S. Heckbert. Hierarchical face clustering on polygonal surfaces. In *Proceedings of the 2001 symposium on Interactive 3D graphics*, pages 49–58, New York, New York, USA, 2001. ACM.

[9] Stuart Geman and Donald Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[10] D D Hoffman and M Singh. Salience of visual parts. *Cognition*, 63(1):29–78, April 1997.

[11] Hossam Isack and Yuri Boykov. Energy-Based Geometric Multi-model Fitting. *International Journal of Computer Vision*, 97:123–147, 2012.

[12] Vladimir Kolmogorov and Ramin Zabih. What Energy functions can be Minimized via Graph Cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159, 2004.

[13] Bao Li, Ruwen Schnabel, Shiyao Jin, and Reinhard Klein. Variational Surface Approximation and Model Selection. *Computer Graphics Forum*, 28(7):1985–1994, October 2009.

[14] Yangyan Li, Xiaokun Wu, Y. Chrysanthou, Andrei Sharf, D. Cohen-Or, and N.J. Mitra. GlobFit: Consistently Fitting Primitives by Discovering Global Relations. *ACM Transactions on Graphics (SIGGRAPH'11)*, 30(4), 2011.

[15] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, Morgan Kaufman, San Mateo, 1988.

[16] Carsten Rother, Vladimir Kolmogorov, Victor Lempitsky, and Martin Szummer. Optimizing Binary MRFs via Extended Roof Duality. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, June 2007.

[17] Ruwen Schnabel, Roland Wahl, and Reinhard Klein. Efficient RANSAC for Point-Cloud Shape Detection. *Computer Graphics Forum*, 26(2):214–226, June 2007.

[18] Vincent Vidal, Christian Wolf, and Florent Dupont. Robust feature line extraction on CAD triangular meshes. In *International Conference on Computer Graphics Theory and Applications*, pages 106–112, Algarve, Portugal, 2011.

[19] Jianhua Wu and Leif Kobbelt. Structure recovery via hybrid variational surface approximation. *Computer Graphics Forum*, 24(3):277–284, September 2005.

[20] D.M. Yan, Yang Liu, and Wenping Wang. Quadric surface extraction by variational shape approximation. *Geometric Modeling and Processing-GMP 2006*, pages 73–86, 2006.

[21] Juyong Zhang, Jianmin Zheng, Chunlin Wu, and Jianfei Cai. Variational Mesh Decomposition. *ACM Transactions on Graphics*, 31(3):21:1—-21:14, August 2012.