Mesh Clustering by Approximating Centroidal Voronoi Tessellation *

Fengtao Fan
Depart. of Computer Science
University of Kentucky
ffan2@uky.edu

Fuhua (Frank) Cheng
Depart. of Computer Science
University of Kentucky
cheng@cs.uky.edu

Conglin Huang
Depart. of Computer Science
University of Kentucky
conglin.huang@uky.edu

Yong Li
Depart. of Computer Science
University of Kentucky
yong.li@uky.edu

Jianzhong Wang
Depart. of Computer Science
University of Kentucky
jwangf@uky.edu

Shuhua Lai
Depart. of Mathematics and
Computer Science
Virginia State University
slai@vsu.edu

ABSTRACT

An elegant and efficient mesh clustering algorithm is presented. The faces of a polygonal mesh are divided into different clusters for mesh coarsening purpose by approximating the Centroidal Voronoi Tessellation of the mesh. The mesh coarsening process after clustering can be done in an isotropic or anisotropic fashion. The presented algorithm improves previous techniques in local geometric operations and parallel updates. The new algorithm is very simple but is guaranteed to converge, and generates better approximating meshes with the same computation cost. Moreover, the new algorithm is suitable for the variational shape approximation problem with $L^{2,1}$ distortion error metric and the convergence is guaranteed. Examples demonstrating efficiency of the new algorithm are also included in the paper.

Categories and Subject Descriptors

I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling —*Physically based modeling*

Keywords

Mesh Clustering, Centroidal Voronoi Tessellation, Shape Approximation

1. INTRODUCTION

3D mesh models are used in many important areas such as geometric modeling, computer animation, and CAD. With the availability of powerful laser scanners, large and dense meshes are easily acquired from physical world. However, since full complexity of such models is not always required, coarsening a dense mesh, i.e., replacing the original mesh with a simpler but close enough mesh, is a necessary preprocessing step in many applications. Many mesh coarsening techniques have been presented, including the global optimization approach [6] and remeshing for mesh coarsening [9, 5].

A third mesh coarsening approach is based on mesh clustering, i.e., partitioning the faces or vertices of the mesh into flat regions (called clusters) which are non-overlapping and connected, and then building an approximating mesh based on the clusters. The clustering process can be done implicitly or explicitly [1]. There are quite a few papers discussing mesh approximation based on explicitly constructing clusters. Clustering by approximating the Centroidal Voronoi Tessellation (CVT) [2] on triangular meshes is first discussed in [15]. After constructing the clusters, the mesh is uniformly coarsened based on the clusters. Adaptive coarsening of a mesh based on clustering from Centroidal Voronoi Tessellation is presented in [17]. An extension from uniform mesh coarsening [15] to anisotropic mesh coarsening is discussed in [16]. A theoretical framework of variational shape approximation based on optimal mesh clustering with respect to some distortion error metric is presented in [1]. Especially, optimal clustering using $L^{2,1}$ metric faithfully captures the anisotropic nature of the mesh. A hierarchical face clustering technique is developed in [12]. Many applications such as collision detection, surface simplification and multiresolution radiosity benefit from this hierarchical clustering technique. Clustering faces in a set of characteristic regions to build a higher-level description of mesh geometry is explored in [8, 14, 11]. Accelerating general iterative clustering algorithms for meshes on GPU is discussed in [7].

This paper is inspired by the work presented in [15, 16, 1]. The goal here is to do clustering by approximating Constrained Centroidal Voronoi Tessellation [3] or Centroidal Voronoi Tessellation [2] on a polygonal mesh. Starting with an initial partitioning of the mesh, the new algorithm iteratively tests the boundary edges between different clusters to update the cluster configuration until the boundary edges do not change any more. This boundary testing algorithm is also discussed in [15, 16]. But we derive a simpler algorithm

*(Does NOT produce the permission block, copyright information nor page numbering). For use with ACM_PROC_ARTICLE-SP.CLS. Supported by ACM.

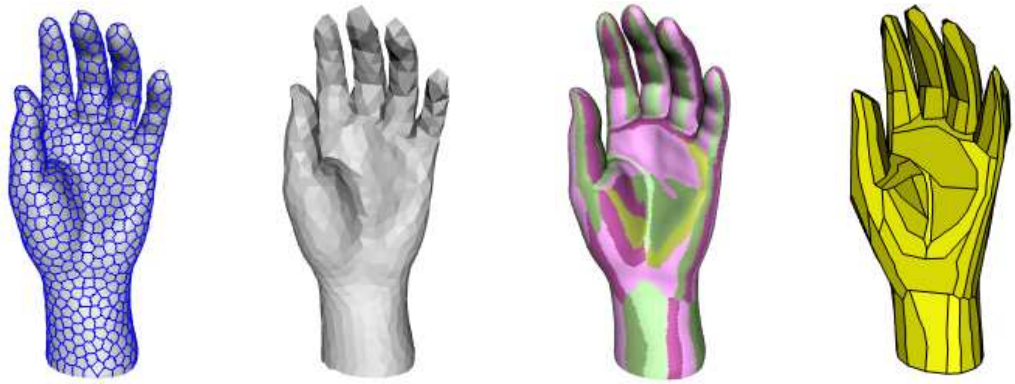

Figure 1: Clustering and approximation results on a hand model: the left-most figure shows the 500 clusters generated by approximating CVT on the mesh; the second from left figure is the uniformly coarsened mesh; the third from left figure has 98 clusters in different colors while using $L^{2,1}$ metric for clustering; the right-most figure is the approximating polygonal mesh.

by presenting a more rigorous mathematical analysis. The new algorithm is intuitive in that it only needs to compare the distances from one face centroid to centroids of adjacent clusters. The new algorithm is also extended for optimal geometric partitioning with respect to $L^{2,1}$ in [1]. The exciting result is that the new algorithm is guaranteed to converge while the algorithm based on Lloyd method in [1] is not.

The remaining part of the paper is organized as follows: Section 2 gives some basics on Centroidal Voronoi Tessellation and its extension; Section 3 presents an analysis and the new clustering algorithms; Section 4 proposes some strategies to make implementation more efficient; test results are shown in Section 5; the conclusion is given in Section 6.

2. CENTROIDAL VORONOI TESSELLATION

Voronoi diagrams or *Voronoi tessellation* are essential structures in computational geometry and have been used in many important applications [13]. Given a domain Ω in $\Re^n$ and a set of points $\{\mathbf{z}_i\}_{i=1}^k$, the corresponding *Voronoi diagram* $\{V_i\}_{i=1}^k$ is a partition of Ω such that:

(1) $V_i \cap V_j = \emptyset$ and $\cup_{i=1}^k \overline{V}_i = \overline{\Omega}$, and
(2) $V_i = \{\mathbf{x} \in \Omega \mid |\mathbf{x} - \mathbf{z}_i| < |\mathbf{x} - \mathbf{z}_j| \text{ for } j = 1, 2, .., k, j \neq i\}$

$\{\mathbf{z}_i\}_{i=1}^k$ are called the *generators* and $\{V_i\}_{i=1}^k$ the *Voronoi regions.*

Centroidal Voronoi Tessellation (CVT) is an extension of Voronoi Tessellation by requiring that the generators are also the mass centroids of the Voronoi regions. Given a density function $\rho(\mathbf{x})$ on V, the mass centroid $\mathbf{z}^*$ of V is defined as

$$\mathbf{z}^* = \frac{\int_V \mathbf{x}\rho(\mathbf{x})d\mathbf{x}}{\int_V \rho(\mathbf{x})d\mathbf{x}}$$

Specifically, CVT of Ω is a minimizer of the energy functional [2] :

$$F(\mathbf{z}) = \sum_{i=1}^{n} \int_{V_i} \rho(\mathbf{x})|\mathbf{x} - \mathbf{z}_i|^2 d\mathbf{x} \qquad (1)$$

where $\mathbf{z}_i \in \Omega$.

Constrained Centroidal Voronoi Tessellation [3] is the restriction of CVT to a surface. If a density function $\rho(\mathbf{x})$ is defined on a surface $\mathbf{S}$, we can define the *constrained mass centroid* $\mathbf{z}^c$ of a region $V \subseteq \mathbf{S}$ as the solution to the following minimization problem:

$$\min_{\mathbf{z} \in \mathbf{S}} \int_V \rho(\mathbf{x})|\mathbf{x} - \mathbf{z}|^2 d\mathbf{x} \qquad (2)$$

A Voronoi Tessellation on a surface $\mathbf{S}$ is a Constrained Centroidal Voronoi Tessellation (CCVT) if and only if the generators $\mathbf{z}_i$ associated with each Voronoi region V_i are also the constrained mass centroid of V_i. Several applications of CCVT can be found in [3]. Furthermore, CCVT of surface $\mathbf{S}$ is also the minimizer of an energy functional similar to the one defined in eq. (1) except now $\mathbf{z}_i \in \mathbf{S}$ [3]. Note that although $\mathbf{x}$ and $\mathbf{z}_i$ are points of the surface $\mathbf{S}$, CCVT uses the Euclidean distance instead of the geodesic distance. Several algorithms for constructing CVT and CCVT, such as the Lloyd method and k-means method, are presented in [2, 3].

In this paper, we will give a rigorous analysis of constructing CCVT on a polygonal mesh. We choose analyzing discrete CCVT because discrete CVT can be viewed as a special case of discrete CCVT. Our derivations are presented below.

3. DISCRETE CONSTRAINED CENTROIDAL VORONOI TESSELLATION ON A POLYGONAL MESH

Given a polygonal mesh M and a cluster number n, we will try to divide the faces of M into n connected sets of faces V_i $(i = 1, 2, \ldots, n)$ by constructing a CCVT on M. These clusters $\{V_i\}$ form a discrete CCVT on the mesh M. Although discrete CCVT can be defined for any polygonal mesh, we will concentrate on triangular meshes in this paper.

In the continuous setting, CCVT is the minimizer of an energy functional similar to the one defined in eq. (1). For the discrete version of CCVT on a triangular mesh M, the region V_i is a connected collection of triangles. We can rewrite

the energy functional as

$$F(\mathbf{z}) = \sum_{i=1}^{n} \Big(\sum_{T_k \in V_i} \int_{T_k} \rho(\mathbf{x}) |\mathbf{x} - \mathbf{z}_i|^2 d\mathbf{x} \Big)$$

where T_k's are triangles in V_i. In this paper, we only consider the uniform case, i.e., $\rho(\mathbf{x}) = 1$. Then the energy functional is

$$F(\mathbf{z}) = \sum_{i=1}^{n} \Big(\sum_{T_k \in V_i} \int_{T_k} |\mathbf{x} - \mathbf{z}_i|^2 d\mathbf{x} \Big) \tag{3}$$

In fact, the following equation holds

$$\int_{T_k} |\mathbf{x} - \mathbf{z}_i|^2 d\mathbf{x} = |\mathbf{x}_k - \mathbf{z}_i|^2 |T_k| + \frac{|T_k|}{12} \sum_{j=1}^{3} |\mathbf{x}_k^j - \mathbf{x}_k|^2 \tag{4}$$

where $|T_k|$ is the area of triangle T_k with vertices $\mathbf{x}_k^j (j = 1, 2, 3)$ and $\mathbf{x}_k$ is the centroid of T_k. This equality is a special case of [10]. In the following, σ_k is used to denote the constant term for triangle T_k. Substituting the integral in eq. (3) with eq. (4), we have

$$F(\mathbf{z}) = \sum_{i=1}^{n} \Big(\sum_{T_k \in V_i} |\mathbf{x}_k - \mathbf{z}_i|^2 |T_k| \Big) + \sum_{T_k \in M} \sigma_k \tag{5}$$

The last constant item is not essential in subsequent work, hence, will be omitted for $F(\mathbf{z})$. The constrained mass centroid $\mathbf{z}_i$ of V_i on a continuous surface $\mathbf{S}$ is defined as a solution to the minimization problem defined in eq. (2) with V replaced with V_i. For discrete CCVT on M, we can use the same argument as in reformulating $F(\mathbf{z})$ to rewrite the minimization problem as:

$$\min_{\mathbf{z} \in \mathbf{M}} \Big(\sum_{T_k \in V_i} |\mathbf{x}_k - \mathbf{z}|^2 |T_k| + \sum_{T_k \in V_i} \sigma_k \Big)$$

The last constant item is not essential in the minimization process and, hence, will be omitted too. Furthermore, the above equation without the constant can be simplified as

$$\min_{\mathbf{z} \in \mathbf{M}} \Big(\sum_{T_k \in V_i} |\mathbf{x}_k - \bar{\mathbf{z}}_i|^2 |T_k| + \sum_{T_k \in V_i} |\bar{\mathbf{z}}_i - \mathbf{z}|^2 |T_k| \Big) \tag{6}$$

where $\bar{\mathbf{z}}_i = \frac{\sum_{T_k \in V_i} |T_k| \mathbf{x}_k}{\sum_{T_k \in V_i} |T_k|}$ is the mass centroid of V_i.

Eq. (6) can be derived by the fact that $\sum_{T_k \in V_i} |T_k| (\mathbf{x}_k - \bar{\mathbf{z}}_i) = 0$. Thus the constrained mass centroid of V_i is the point on M that is closest to its mass centroid $\bar{\mathbf{z}}_i$. Eqs. (5) and (6) are the counterparts of eqs. (1) and (2) in the discrete case. Before we describe the algorithm, two important properties have to be highlighted first.

PROPERTY 3.1. *Let $\{(V_i, \mathbf{z}_i)\}$ be the current cluster configuration where $\mathbf{z}_i$ is the constrained mass centroid of V_i, and for each triangle $T_r \in V_i$'s, let $\mathbf{x_r}$ be its centroid. If $|\mathbf{x_k} - \mathbf{z}_q|^2 < |\mathbf{x_k} - \mathbf{z}_p|^2$ for some triangle $T_k \in V_p$ and V_q adjacent to V_p, then*

$$F'(\mathbf{z}) < F(\mathbf{z})$$

where

$$F'(\mathbf{z}) = \sum_{i=1}^{n} \Big(\sum_{T_k \in V_i'} |\mathbf{x}_k - \mathbf{z'}_i|^2 |T_k| \Big), \tag{7}$$

$\mathbf{z'}_i$ is the constrained mass center of V_i' and

$$V_i' = \begin{cases} V_i & i \neq p, q \\ V_p - \{T_k\} & i = p \\ V_q \cup \{T_k\} & i = q \ . \end{cases}$$

Note that since $|\mathbf{x_k} - \mathbf{z}_q|^2 < |\mathbf{x_k} - \mathbf{z}_p|^2$, it is clear that

$$\sum_{T_j \in V_p - \{T_k\}} |\mathbf{x}_j - \mathbf{z}_p|^2 |T_j| + \sum_{T_j \in V_q \cup \{T_k\}} |\mathbf{x}_j - \mathbf{z}_q|^2 |T_j| < \sum_{T_j \in V_p} |\mathbf{x}_j - \mathbf{z}_p|^2 |T_j| + \sum_{T_j \in V_q} |\mathbf{x}_j - \mathbf{z}_q|^2 |T_j|.$$

From the minimization property of the constrained mass centroid $\mathbf{z'}_i$, the following inequality holds:

$$\sum_{T_j \in V_t'} |\mathbf{x}_j - \mathbf{z'}_t|^2 |T_j| \leq \sum_{T_j \in V_t'} |\mathbf{x}_j - \mathbf{z}_t|^2 |T_j| \ , \quad t = p, q$$

Combining these two steps, $F'(\mathbf{z}) < F(\mathbf{z})$ follows readily.

PROPERTY 3.2. *Let $\{(V_i, \mathbf{z}_i)\}$ be the current cluster configuration, and triangles $T_k \in V_p$ and $T_s \in V_q$ with centroids $\mathbf{x}_k$ and $\mathbf{x}_s$, respectively, share a common edge. If $|\mathbf{x}_k - \mathbf{z}_p|^2 > |\mathbf{x}_k - \mathbf{z}_q|^2$, $|\mathbf{x}_s - \mathbf{z}_q|^2 > |\mathbf{x}_s - \mathbf{z}_p|^2$ and $|\mathbf{x}_k - \mathbf{z}_p|^2 |T_k| + |\mathbf{x}_s - \mathbf{z}_p|^2 |T_s| < |\mathbf{x}_k - \mathbf{z}_q|^2 |T_k| + |\mathbf{x}_s - \mathbf{z}_q|^2 |T_s|$ then*

$$F'(\mathbf{z}) < F(\mathbf{z})$$

where $F'(\mathbf{z})$ is defined in eq. (7).

This property can easily be proved following an argument similar to that of property 3.1. In fact, reassigning either T_k or T_s will lower the value of the energy functional $F(\mathbf{z})$. In a greedy spirit, we simply choose the smaller one, which is reflected by the third given inequality. One can not simply assign T_k to V_q and T_s to V_p because the result could violate the connectivity requirement for clusters.

3.1 Energy minimization

Recall that a discrete CCVT of a triangular mesh M is a minimizer of the discrete energy functional (5). In the following we propose an algorithm to iteratively reduce the value of $F(\mathbf{z})$ until a limit point is reached. The main idea of the algorithm is to update the clusters by comparing distances from triangle centroids of a cluster to mass centroids of adjacent clusters. The triangles that have to be considered are just boundary triangles, i.e., triangles sharing a *cluster edge*. A mesh edge is called a *cluster edge* if it is shared by two triangle faces of different clusters. The distance comparing procedure is stated below.

Let edge e_{lr} be a *cluster edge* in the current cluster configuration $\{(V_i, \mathbf{z}_i)\}$. e_{lr} is shared by triangles T_l and T_r, where $T_l \in V_p$ and $T_r \in V_q$ are in different clusters. Let $\mathbf{x}_l$ and $\mathbf{x}_r$ be the centroids of T_l and T_r, respectively. Denote $|\mathbf{x}_l - \mathbf{z}_p|^2$, $|\mathbf{x}_l - \mathbf{z}_q|^2$, $|\mathbf{x}_r - \mathbf{z}_p|^2$ and $|\mathbf{x}_r - \mathbf{z}_q|^2$ with d_{lp}, d_{lq}, d_{rp} and d_{rq}, respectively. We need to compare d_{lp} with d_{lq}, and d_{rp} with d_{rq}, totally four cases. Figure 2 illustrates these 4 cases.

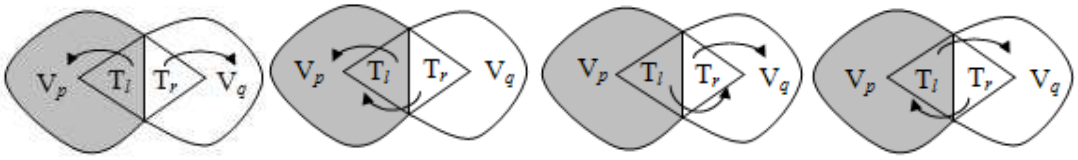

Figure 2: Illustration of 4 cases in distance comparison. The presence of an arrow indicates direction of the movement after the comparison. These are cases 1, 2, 3 and 4 from left to right in that order.

1. $d_{lp} \leq d_{lq}$ and $d_{rp} \geq d_{rq}$.
   Do nothing. This is exactly what the convergent state should be.

2. $d_{lp} \leq d_{lq}$ and $d_{rp} < d_{rq}$.
   Move T_r to V_p. According to property 3.1, this movement lowers the value of the energy functional $F(\mathbf{z})$.

3. $d_{lp} > d_{lq}$ and $d_{rp} \geq d_{rq}$.
   Move T_l to V_q. The new value of the energy functional $F(\mathbf{z})$ will be lower, according to property 3.1.

4. $d_{lp} > d_{lq}$ and $d_{rp} < d_{rq}$.
   One more test is needed to decide which triangle should be moved.

   - If $d_{lp}|T_l| + d_{rp}|T_r| < d_{lq}|T_l| + d_{rq}|T_r|$, move T_r to V_p.
   - Otherwise, move T_l to V_q.

   The value of the energy functional $F(\mathbf{z})$ will be lower after the movement, according to property 3.2.

Instead of updating the mass centroid of each cluster immediately after distance comparison for each *cluster edge*, it is better to do updating after the distance comparison for all the *cluster edges*. We call such a scheme *configuration-wise updating*. Its correctness follows from the optimality of the constrained mass centroid and properties 3.1 and 3.2.

With a valid initial cluster configuration, we perform distance comparison for each *cluster edge*. After completing the distance comparison process for all *cluster edges*, we update the mass centroids of clusters and update the *cluster edge* set. This process is iterated until the *cluster edge* set no longer changes.

It is obvious that the energy functional $F(\mathbf{z})$ has a global minimum on the triangular mesh M. As $F(\mathbf{z})$ decreases strictly after each configuration-wise updating, it is guaranteed to converge to a limit point. The "minimum" it achieves might not be the global minimum of $F(\mathbf{z})$. For our clustering goal, this doesn't matter much. The limit cluster configuration always gives a very good clustering of M.

Remark: Although our results are for discrete CCVT on M, there are parallel results for discrete CVD on $M \subset \Re^3$. Because the mass centroid $\bar{\mathbf{z}}_i = \frac{\sum_{T_k \in V_i} |T_k| \mathbf{x}_k}{\sum_{T_k \in V_i} |T_k|}$ of a cluster V_i on a triangular mesh M can also be viewed as a solution to the minimization problem

$$\min_{\mathbf{z} \in \Re^3} \sum_{T_k \in V_i} |\mathbf{x}_k - \mathbf{z}|^2 |T_k|$$

This is true because we have

$$\sum_{T_k \in V_i} |\mathbf{x}_k - \mathbf{z}|^2 |T_k| = \sum_{T_k \in V_i} |\mathbf{x}_k - \bar{\mathbf{z}}_i|^2 |T_k| + \sum_{T_k \in V_i} |\bar{\mathbf{z}}_i - \mathbf{z}|^2 |T_k|$$

Thus it is obvious that $\bar{\mathbf{z}}_i$ is the solution to this minimization problem. We present the approximated results by CVT on the bunny model in Figure 3. The discrete CVT method runs much faster than the discrete CCVT method because the CCVT method needs to find the closest points in each iteration. Our examples in this paper are mainly the results from the discrete CVT method.

3.2 Boundary testing algorithm for clustering with $L^{2,1}$ metric

A novel metric $L^{2,1}$ for geometric partitioning of a triangular mesh M is well studied in [1]. The optimal geometric partition of M for a given partition number n can be defined as the minimizer of the distortion error:

$$E(M, P) = \sum_{i=1}^{n} \sum_{T_k \in R_i} |\mathbf{n}_k - \mathbf{N}_i|^2 |T_k|$$

where R_i's are connected collections of triangles, $\mathbf{n}_k$ is the unit normal of triangle T_k and $\mathbf{N}_i$ is the normalized vector of $\sum_{T_k \in R_i} \mathbf{n}_k |T_k|$.

This functional is very similar to the cost functional for CCVT. In fact, there are parallel results for this distortion error due to the optimality of $\mathbf{N}_i$. Precisely, $\mathbf{N}_i$ is the solution to the minimization problem:

$$\min_{|\mathbf{N}|=1} \sum_{T_k \in R_i} |\mathbf{n}_k - \mathbf{N}|^2 |T_k|$$

Note that, similar to eq. (6), we have

$$\sum_{T_k \in R_i} |\mathbf{n}_k - \mathbf{N}|^2 |T_k| = \sum_{T_k \in R_i} |\mathbf{n}_k - \overline{\mathbf{N}}_i|^2 |T_k| + \sum_{T_k \in R_i} |\overline{\mathbf{N}}_i - \mathbf{N}|^2 |T_k|$$

where $\overline{\mathbf{N}}_i = \frac{\sum_{T_k \in R_i} \mathbf{n}_k |T_k|}{\sum_{T_k \in R_i} |T_k|}$. And it is obvious that

$$|\mathbf{N}_i - \overline{\mathbf{N}}_i|^2 = \min_{|N|=1} |\mathbf{N} - \overline{\mathbf{N}}_i|^2$$

Thus the minimization property of $\mathbf{N}_i$ follows. With the same arguments as above, one can design a similar *configuration-wise updating* algorithm for $L^{2,1}$ metric which is guaranteed to converge.

4. IMPLEMENTATION

Several strategies can be used to accelerate the clustering process. Our iterative algorithm always begins with a valid initial cluster configuration, i.e., the clusters are connected

Models	#F (org)	#V (org)	#V (approx)	time(s)	min $\angle$	Ave. min $\angle$	$\angle < 30^\circ$	$Q_{\min}$	Q_{ave}
hand	72.9k	36.6k	1000	0.102	31.177	49.5895	0	0.580444	0.868164
bunny	69.4k	34.8k	500	0.095	31.3123	50.321	0	0.581436	0.879907
statue	272k	136k	800	1.457	32.0611	51.9163	0	0.541623	0.900297

Table 1: Results for uniform mesh coarsening

and non-overlapping. We apply the hierarchical face clustering idea in [12] to design our cluster initialization. Hierarchical face clustering respects the connected requirement of clusters strictly and builds such a hierarchical structure on the dual graph of the mesh. It then applies edge contraction on the dual graph iteratively. The edge chosen for contraction is based on a cost function. In [12], the cost function is the planarity criterion. For our algorithm, we define the cost function for an edge e_{ij} that connects $(V_i, \mathbf{z}_i, |V_i|)$ and $(V_j, \mathbf{z}_j, |V_j|)$ as:

$$F(e_{ij}) = \frac{|\mathbf{z}_{ij} - \mathbf{z}_i|^2 |V_i| + |\mathbf{z}_{ij} - \mathbf{z}_j|^2 |V_j|}{|V_i| + |V_j|}$$

where $|V_p| = \sum_{T_k \in V_p} |T_k|$ $(p = i, j)$, $\mathbf{z}_i$ and $\mathbf{z}_j$ are the "mass centroids" of V_i and V_j, respectively, and $\mathbf{z}_{ij} = \frac{\mathbf{z}_i|V_i| + \mathbf{z}_j|V_j|}{|V_i| + |V_j|}$. $\mathbf{z}_i$ depends on our distortion error metric. It is the mass centroid for the CVT constructing case and is the unit normal of the proxy plane for the optimal geometric partitioning case. The edge cost function $F(e_{ij})$ is just the energy functional $F(\mathbf{z})$ when there are only two faces V_i and V_j. Our hierarchical initialization can generates exactly n connected clusters and accelerates the clustering convergence.

Another accelerating strategy is to keep tracking whether a cluster is about to settle down. This is useful because only a few cluster edges need to be updated as the algorithm proceeds to convergence. One more issue is the validity of clusters. As stated before, a valid cluster must be connected, but our algorithm does not guarantee that the resulting clusters are connected. When such case happens, reassignment of clusters are performed .

Depending on the clustering criteria, a mesh M can either be uniformly coarsened after the construction of a discrete CVT or CCVT, or be approximated in an anisotropic fashion following the construction of an optimal geometric partition with respect to an $L^{2,1}$ metric. Uniform mesh coarsening and anisotropic approximation for $L^{2,1}$ metric are discussed in details in [15, 16] and [1], respectively.

5. RESULTS

The algorithms presented in this paper are implemented on a laptop computer with 1G memory and Intel Core 2 CPU T7200 under Windows. Performance data for uniform mesh coarsening applications are collected in Table 1. Experiments for anisotropic shape approximation are also carried out. Notice that the new algorithms run very fast. It takes only a few seconds to get the job done for a mesh with more than 200k faces.

For the uniform mesh coarsening application, the quality of the output mesh M is measured in several aspects, as listed in Table 1. 'min $\angle$' stands for the minimum angle degree of the triangle faces in M. Similarly, 'Ave. min $\angle$' computes the average minimum angle degree. '$\angle < 30^\circ$' counts the number of angles smaller than 30 degrees. '$Q_{\min}$' (minimal quality) and 'Q_{ave}' (average quality) measure the triangle shapes. Both terms are defined in [4]. The examples have also been tested using the program provided by the authors of [15, 16]. The execution times for models bunny and hand are 0.328s and 0.266s, respectively, which are slower than the new algorithm. However, the execution times on the statue modelis 0.954s, which is better than the new algorithm. But the output meshes of the new algorithms always have better mesh quality.

The new algorithm for anisotropic approximation with $L^{2,1}$ metric runs very fast and gives good approximation results. It takes 1.542s to generate 98 clusters on the hand model, 0.032s for 32 clusters on the fandisk model and 1.8327s for 120 clusters on the monster model. This shows that the new algorithm is practical. The new algorithm contracts one face for each cluster. The anisotropic nature of the $L^{2,1}$ error metric is demonstrated in these examples.

6. CONCLUSIONS

In this paper we propose a novel clustering algorithm for a polygonal mesh M by approximating CVT or CCVT on M. The new clustering algorithm is also suitable for clustering construction with respect to the $L^{2,1}$ error metric. We present a rigorous mathematical analysis for the new algorithm. Our algorithm possesses the intrinsic distance comparison as the local geometric operation, which is simpler and more intuitive than those used in [15, 16]. Moreover, our algorithm updates the cluster configuration only after comparing all *cluster edges*. The proposed algorithm based on Lloyd method for constructing the optimal geometric partition in [1] is not guaranteed to converge. But the new algorithm is proved to converge for constructing discrete CCVT and CVT on M or clustering with $L^{2,1}$ metric. Although the new algorithm runs more or less as those in [16], the coarse mesh produced by the new algorithm has a better mesh quality. Depending on the clustering criteria, we show examples for both isotropic and anisotropic mesh approximations. The anisotropic mesh approximation by using CVT is also investigated in [16]. It seems to be an interesting problem to generalize the new algorithm for the anisotropic case. This will be investigated in the future.

7. ACKNOWLEDGMENTS

We thank anonymous reviewers for their valuable comments.

8. REFERENCES

[1] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. *ACM Transactions on Graphics. Special issue for SIGGRAPH conference*, pages 905–914, 2004.

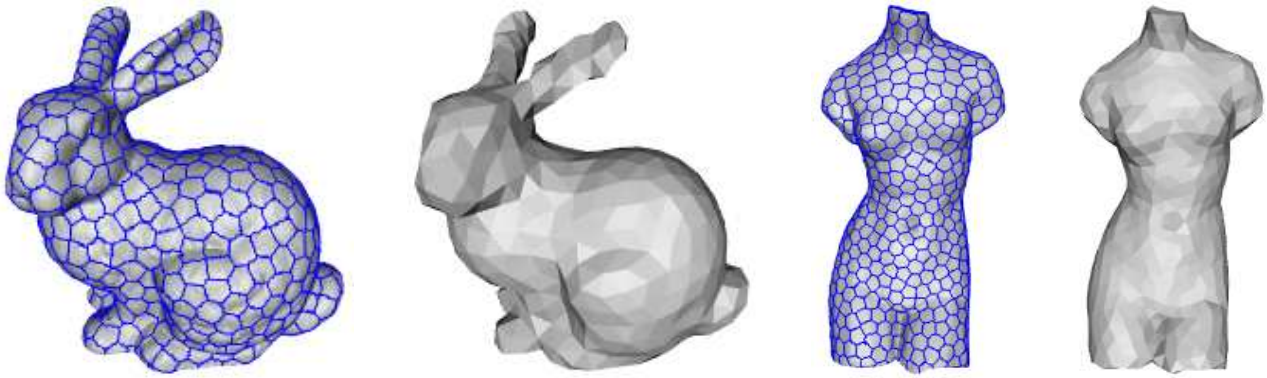

Figure 3: The bunny model with 500 clusters and its coarsened mesh by CVT; The statue model with 800 clusters and the coarsened mesh.

[2] Q. Du, V. Faber, and M. Gunzburger. Centroidal voronoi tessellations: Applications and algorithms. *SIAM Rev.*, 41(4):637–676, 1999.

[3] Q. Du, M. D. Gunzburger, and L. Ju. Constrained centroidal voronoi tessellations for surfaces. *SIAM J. Sci. Comput.*, 24(5):1488–1506, 2002.

[4] P. Frey and H. Borouchaki. Surface mesh evaluation. In *16th International Meshing Roundtable*, pages 363–374, 1997.

[5] I. Guskov, K. Vidimče, W. Sweldens, and P. Schröder. Normal meshes. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 95–102, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[6] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *ACM SIGGRAPH Proceedings*, pages 19–26, 1993.

[7] J. C. H. Jesse D. Hall. Gpu acceleration of iterative clustering. *The ACM Workshop on General Purpose Computing on Graphics Processors, and SIGGRAPH 2004 poster*, Aug. 2004.

[8] A. Kalvin and R. Taylor. Superfaces: Polygonal mesh simplification with bounded error. *IEEE Comput. Graph. Appl.*, 16(3):64–77, 1996.

[9] L. Kobbelt, J. Vorsatz, U. Labsik, and H.-P. Seidel. A shrink wrapping approach to remeshing polygonal surfaces. 18(3):119–130, 1999. (Proceedings of EUROGRAPHICS'99).

[10] J. B. Lasserre and K. E. Avrachenkov. The multi-dimensional version of $\int_a^b x^p dx$. *The American Mathematical Monthly*, 108(2):151–154, 2001.

[11] B. Lévy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph.*, 21(3):362–371, 2002.

[12] A. W. M. Garland and P. S. Heckbert. Hierarchical face clustering on polygonal surfaces. In *Proceedings of the Symposium on Interactive 3D Graphics*, 2001.

[13] A. Okabe, B. Boots, and K. Sugihara. *Spatial Tessellations Concepts and Applications of Voronoi Diagrams*. John Wiley & Son, 1992.

[14] A. Sheffer. Model simplification for meshing using face clustering. *Comptuer-Aided Design*, 33:925–934, 2001.

[15] S. Valette and J.-M. Chassery. Approximated centroidal voronoi diagrams for uniform polygonal mesh coarsening. 23(3):381–389, 2004. (Proc. Eurographics'04).

[16] S. Valette, J.-M. Chassery, and R. Prost. Generic remeshing of 3d triangular meshes with metric-dependent discrete voronoi diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 10(2):369–381, 2008.

[17] S. Valette, I. Kompatsiaris, and J.-M. Chassery. Adaptive polygonal mesh simplification with discrete centroidal voronoi diagrams. In *Proceedings of 2nd International Conference on Machine Intelligence ICMI 2005*, pages 655–662, 2005.

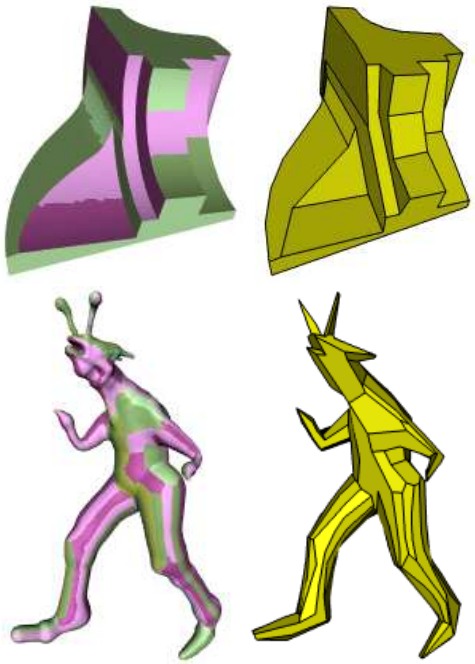

Figure 4: A fandisk model with 32 clusters and its approximating mesh; A monster model with 120 clusters and its approximating mesh.