The Normal Map Based on Area-Preserving Parameterization

Hui Zhao* Kehua Su† Ming Ma‡ Na Lei.§ Li Cui¶ Xianfeng Gu∥

Figure 1: (a) High resolution mesh; (b) simplified mesh; (c) rendered by normal map; (d) conformal results; (e) OMT result; (f) our normal map texture

Abstract

In this paper, we present an approach to enhance and improve the current normal map rendering technique. Our algorithm is based on semi-discrete Optimal Mass Transportation (OMT) theory and has a solid theoretical base. The key difference from previous normal map method is that we preserve the local area when we unwrap a disk-like 3D surface onto 2D plane. Compared to the currently used techniques which is based on conformal parameterization, our method does not need to cut a surface into many small pieces to avoid the large area distortion. The following charts packing step is also unnecessary in our framework. Our method is practical and makes the normal map technique more robust and efficient.

Keywords: power Voronoi diagram, power Delaunary triangulation, optimal mass transportation, area preserving

Concepts: • Computing methodologies → Mesh geometry models;

1 Introduction

The normal map [Cignoni et al. 1998] is an important and crucial rendering technique in computer graphics. This traditional method is used widely to display a model efficiently in past twenty years. When we render a shape with very high details, we need a dense mesh to represent the geometry of a shape. Normally this kind of meshes have more than ten thousand faces, and can be obtained by sculpturing with some softwares, for example, “Zbrush”, or by scanning from real objects.

In real time rendering, the color of every pixel in the rendering image is determined by the normal of its corresponding vertex and lighting. When these dense models are displayed, the rendering is very time costing due to the large number of triangle faces. If we simplify the model to speed up the rendering time, the local details of the shape will be lost. The normal map technique solves the problem by storing the vertex normals of original dense mesh into a 2D texture image. When we render the simplified meshes, the light shading is calculated in pixel space instead of vertex space, and the normal directions of pixels are determined by the values stored in the normal map texture instead of computing from the normals of the vertice of the simplified models. Therefore, in Figure 1, the rendered picture (c) of the combination of simplified mesh (b) and normal map texture (f) looks similar to the original dense mesh (a), but it is much more efficient in rendering speed.

The conformal parameterization preserves the angle, therefore it is suitable for texturing applications. In normal map, the area-preserving parameterization is an ideal tool. In this paper, we propose to use an area-preserving mesh parameterization algorithm to unwrap the original high resolution 3D mesh, therefore we can guarantee that uniform sampling in 2D will lead to a uniform sampling in 3D. Our method is based on solid discrete optimal mass transportation(OMT) theory, and it can avoid the cutting step and packing step used in previous method. In traditional method, the normal map texture depends on both simplified mesh and high resolution mesh. In ours, the texture depends on the high resolution mesh only. This feature makes our method more flexible and insensitive to the simplification step that is used in previous methods. Our contribution is two sides, one is proposing a different work flow to generate normal map textures; the second one is applying discrete OMT in our normal map work flow to generate the uniform sampling and simplified meshes.

2 Previous Work

The discussion of normal map and mesh parameterization is vast. In this section, we only briefly review the most related works to our current work.

Normal Map. The idea of rendering using normal map with simplified meshes was proposed in [Krishnamurthy and Levoy 1996]. The original dense irregular meshes were converted into tensor product
B-spline surface patches with a displacement maps. For transferring details with normal maps from high resolution to low resolution polygon meshes, they presented texture deviation metric in [Cignoni et al. 1998]. At the same year, for this purpose, the authors in [Cignoni et al. 1998] proposed a general method for preserving detail on simplified meshes with 2D textures. Guskov et al. proposed normal meshes with multiresolution capability to represent geometry in [Guskov et al. 2000]. In [Gu et al. 2002; Snyder et al. 2003], a geometry image approach is presented to store normal efficiently.

There is a lot of mesh parameterization methods [Hormann and Greiner 2000; Sander et al. 2002; Floater 2003; Desbrun et al. 2002; Sheffer et al. 2005; Weber and Zorin 2014]. For in-depth survey, we refer the readers to [Sheffer et al. 2006; Sheffer et al. 2007]. A method which preserves the local area as much as possible is proposed in [Liu et al. 2008]. Our method preserves the local area accurately. In [Kovalsky et al. 2016; Fu et al. 2015; Fu and Liu 2016; Rabinovich et al. 2017; Smith and Schaefer 2015], some injective and bijective mesh parameterization approaches are discussed. These methods do not aim at area preserving, therefore they are not suitable for our normal mapping application.

**Optimal Mass Transportation (OMT).** Monge first raised the classical Optimal Mass Transport Problem [Monge 1781; Bonnotte 2013]. Moving a pile of soil from one place to another, OMT can be solved by Monge-Kantorovich theory. The Monge problem was generalized by Kantorovich [Kantorovich 1942] to an optimal transport plan based on linear programming. For a special convex function, the solution is unique and the optimal transport plan problem can be solved with various efficiency. In recent years, a practical algorithm was proposed in [Lévy 2015] to compute the optimal transport map by using a quasi-Newton method. The hierarchical optimization greatly improves the efficiency. It makes a great development in computation of optimal mass transportation.

OMT is also used in a variety of practical applications: such as blue noise [de Goes et al. 2012]; 2D shape simplification [De Goes et al. 2011], image registration [Haker et al. 2004], image retrieval [Rubner et al. 2000]; image segmentation [Rabin and Papadakis 2015], geodesic distances on 3D meshes [Solomon et al. 2014], shape interpolation [Solomon et al. 2015], displacement interpolation [Bonneel et al. 2011], histogram regression [Bonneel et al. 2016] and shape matching [Su et al. 2015].

Our algorithm relies heavily on the computational geometry techniques of power diagram [Aurenhammer 1987] and weighted Delaunary triangulation. Power diagram is also used in fluid simulation [De Goes et al. 2015], self-supporting structure [De Goes et al. 2013; Liu et al. 2013], weighted triangulation [Goes et al. 2014], animating bubble [Busaryev et al. 2012] and generalized barycentric coordinates [Budynski et al. 2016].

Recently, based on the theoretic work of [Gu et al. 2016], which solves the discrete optimal mass transportation map by convex optimization and gives an explicit geometric interpretation of the Hessian matrix of the convex energy, several applications are developed and applied, such as area preserving [Zhao et al. 2013; Su et al. 2016c], volume preserving mapping [Su et al. 2016a; Su et al. 2016b] and brain morphological study [Su et al. 2013]. Our approach relies on this discrete OMT theory as well.

### 3 Theoretic Background

In the interest of being self-contained, in this section, we review some basic facts of discrete optimal mass transportation, which are the theoretic foundation of our method. The more mathematic details and proofs are explained in [Gu et al. 2016]. At first we introduce the problem in smooth case briefly, then we will discuss its discrete counterpart.

**Smooth OMT.** In 18th century, Monge [Bonnotte 2013] proposed the optimal mass transportation problem, which seeks a minimal transportation cost. Denote $X$ and $Y$ as two metric spaces with probability measures $\mu$ and $\nu$ respectively, and $X$ and $Y$ have equal total measures, i.e. $\int_X \mu = \int_Y \nu$. For any measurable set $B \subseteq Y$, it satisfies $\int_{B^{-1}(B)} \mu = \int_B \nu$, then the map $T : X \rightarrow Y$ is measure preserving. If this condition is satisfied, we use $v = T \mu$ to denote the push forward measure of $\mu$ that is induced by $T$. The transportation cost from $x \in X$ to $y \in Y$ is represented as $c(x,y)$, then the total transportation cost of $T$ can be computed by the following:

$$E(T) := \int_X c(x,T(x))d\mu(x).$$

In our application, we are interested in that the transportation cost is the quadratic Euclidean distance, then the following theorem holds:

**Theorem 1** [Brenier 1991] Suppose $X, Y$ are subsets in $\mathbb{R}^n$, the source $X$ is a convex domain, the transportation cost is the quadratic Euclidean distance, $c(x,y) = |x-y|^2$. Given probability measures $\mu$ and $\nu$ on $X$ and $Y$ respectively, then there is a unique optimal transportation map $T : (X, \mu) \rightarrow (Y, \nu)$, furthermore there is a convex function $f : X \rightarrow \mathbb{R}$, unique up to a constant, and the optimal mass transportation map is given by the gradient map $T : x \mapsto \nabla f(x)$.

When $f$ is with second order smoothness, i.e. $f \in C^2(X, \mathbb{R})$ and the measures $\mu$ and $\nu$ are also smooth, then $f$ satisfies the famous Monge-Ampère equation that is highly non-linear and has the following form:

$$\det \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} \end{pmatrix} = \frac{\mu(x_1, x_2)}{\nu \circ \nabla f(x_1, x_2)}.$$

**Discrete OMT.** We call optimal mass transportation problem as semi-discrete OMT, when the source domain is continuous and the target domain consists of discrete points. Semi-discrete OMT has a nice geometric interpretation that we illustrate in Figure 2. We can solve this problem using variational approach by the convex optimization method.

We summarize the geometric explanation of semi-discrete OMT in the following:

1. The target space $T$ is discretized into $T = \{q_1, q_2, \cdots, q_k\}$ with Dirac measure $\nu = \sum_{i=1}^k \delta(q - q_i)$, as shown in , the lower right picture of Figure 2.
2. A height vector which consists of $k$ real numbers is denoted as $h = (h_1, h_2, \cdots, h_k) \in \mathbb{R}^k$.
3. For each $q_i \in T$, a hyperplane is defined on $V$:

$$\pi_i(h) : (x, q_i) + h_i = 0,$$

where $(,)$ is the inner product in $\mathbb{R}^d$, which is shown as the red or blue polygon in the upper left picture of Figure 2.
4. A piece-wise linear function is defined as:

\[ u_h(x) = \max_{1 \leq i \leq k} \{ (x, q_i) + h_i \}. \]

The upper envelope \( \mathcal{E}(h) \) of the planes \( \{ \pi_i(h) \} \) can be represented by the upper envelope \( \mathcal{E}(h) \) of the planes \( \{ \pi_i(h) \} \). For example, the left top picture of Figure 2.

5. The projection of \( \mathcal{E}(h) \) leads to the power Voronoi decomposition \( \mathcal{V}(h) \) of source domain \( \Omega \)

\[ \Omega = \bigcup_{i=1}^k W_i(h), \quad W_i(h) := \{ x \mid (x, q_i) + h_i \geq (x, q_j) + h_j + j \} \cap \Omega. \]  

Each cell \( W_i(h) \) is the corresponding projection face, namely \( W_i = \{ x \mid \nabla \mu_i(x) = q_i \} \). For example, the red and blue polygons in the left bottom picture of Figure 2.

6. Denote the coordinates of \( q_i \in \mathbb{R}^2 \) as \((x_i, y_i)\), then for each hyperplane \( \pi_i(h) \), its corresponding dual point \( \pi^*_i(h) \in \mathbb{R}^3 \) is constructed as follows:

\[ \pi^*_i(h) = (x_i, y_i, -h_i), i = 1, 2, \cdots, k. \]

7. The upper envelope \( \mathcal{E}(h) \) of \( \{ \pi_i(h) \} \) is the dual to the convex hull \( C(h) \) of \( \{ \pi^*_i(h) \} \).

8. The convex function \( \eta_h \) on each cell \( W_i(h) \) is a linear function \( \pi_i(h) \), therefore, the gradient map \( \nabla \eta_h : W_i(h) \rightarrow q_i, i = 1, 2, \cdots, k \) maps each cell \( W_i(h) \) to a single point \( q_i \). For example, the right picture of Figure 2.

9. The projection of \( C(h) \) induces the weighted Delaunay triangulation \( \mathcal{T}(h) \) of the discrete samples \( \{ q_i \} \).

10. The gradient map of the convex function \( \nabla \eta_h \) maps each power Voronoi cell \( W(h) \) to a sample point \( q_i \).

11. We define the OMT energy \( E(h) \) as the following:

\[ E(h) = \int_{\Omega} \eta_h(x) p(x) dx - \sum_{i=1}^k W_i h_i. \]  

The first item of this OMT energy \( E(h) \) has a geometric meaning: it is the volume of the convex polyhedron bounded by the graph \( G(h) \) and the cylinder through the boundary of \( \Omega \), as shown in Figure 3.

12. The gradient of the energy is computed by:

\[ \nabla E(h) = \int_{W_i(h)} \mu - v_i. \]  

13. The Hessian of \( E(h) \) is given by:

\[ \frac{\partial^2 E(h)}{\partial h_i \partial h_j} = \begin{cases} \frac{f_{ij}(h)}{|q_i - q_j|^2} & W_i(h) \cap W_j(h) \cap \Omega \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \]  

where the cells \( W_i(h) \) and \( W_j(h) \) have a common face \( f_{ij}(h) = W_i(h) \cap W_j(h) \cap \Omega \).

Figure 2: The geometric interpretation of semi-discrete OMT.

Figure 3: The geometric interpretation of the first item of the OMT energy.

It is proved in [Gu et al. 2016] that the admissible space \( H \) of height vectors \( h \) is convex, and the Hessian matrix is positive definite on \( H \), then the energy \( E(h) \) is convex. The existence and uniqueness of the global minimum is also proved in [Gu et al. 2016] and can be obtained efficiently using Newton’s method.

4 Our Work Flow

The traditional normal map texture production technique is very mature and widely used in industry software, such as the software Melody [Melody 2004]. Their work flow is as follows: the following: (a) the original high resolution mesh (Figure 5a) is simplified into a low resolution mesh (Figure 5b) using some simplification algorithm, for example the one in [Garland and Heckbert 1997]; (b) the simplified low resolution mesh is cut into some small pieces; (c) these pieces are unwrapped into the 2D flat meshes; (d) these small 2D meshes are packed into a rectangle; (e) the rectangle is sampled with a uniform grid; (f) these 2D grids points are mapped onto their corresponding 3D points inside the faces of the simplified mesh; (g) the 3D points are ray casted into original high resolution mesh to get their normal directions; (h) the \((x, y, z)\) coordinates of the normal directions is converted into \((R, G, B)\) value of the color and stored into a picture (Figure 5c).

Traditional normal map texture creation flow is complicated. In this paper, we propose a novel and simple work flow for the normal map texture generation and the simplified mesh creation. Traditionally the simplification and the texture production algorithms are independent with each other. In our approach, we combine them into a single component. Our work flow is as follows:

1. We unwrap the original high resolution mesh into a 2D disk (or rectangle).
2. The disk is sampled uniformly with a grid of high resolution.
3. The grid points are mapped to the original high resolution mesh to achieve the normal map texture.
4. The same disk is sampled uniformly with a second different grid of low resolution.
5. These grid points are mapped again to the high resolution mesh to obtain the simplified mesh.

To be successful in normal mapping rendering, the normal sampling should be uniform. In traditional method, to achieve uniform sampling, the simplified mesh is cut into small pieces to avoid large area distortion (Figure 5c). Our method does not need to cut the mesh and pack the pieces (Figure 5d), therefore our method is much more space efficient. The key factor in the normal map techniques is unwrapping a 3D mesh onto the 2D plane, a good unwrapping algorithm is crucial to the success of the normal map technique. The commonly used unwrapping algorithms are based on the conformal mesh parameterization, which preserves the angle, but it leads to large area distortion. Although the grid sampling in unwrapped 2D plane is uniform, the corresponding points in 3D mesh is not uniform due to the area distortion. In Figure 6, the 3D hulk mesh (Figure 6a) is parameterized conformally onto a 2D disk (Figure 6b). The hulk’s head is mapped into a very small area. If we sample the disk uniformly, we get a very small number of corresponding sampling in 3D hulk’s head. To solve the problem, normally the meshes are cut into a lot of small pieces to decrease the area distortion. In Figure 6c, the disk is obtained by our area-preserving parameterization, therefore the corresponding sampling in 3D hulk is also uniform.

Instead of global area, our goal is to preserve the local area, i.e. the area of the neighborhood of every vertex. There are infinite solutions for the local area-preserving parameterization. We use the semi-discrete OMT to single out a special one, such that the problem can be solved by a practical, fast algorithm.

5 Our Algorithm

In the first step, we resize the original 3D mesh by the area of a unit disk (or a unit square). In the second step, we map the 3D mesh onto a unit disk by harmonic map (or conformal map) with the algorithms in [Gortler et al. 2006; Gu and Yau 2003]. The map result keeps the global area unchanged, but it is not local area-preserving. We apply the semi-discrete OMT to adjust the conformal map to the local area-preserving map.

The upper envelop, convex hull and the weighted Delaunay triangulation, shown in Figure 2, are used to explain the geometric meaning of semi-discrete OMT tool. In our algorithm, we only need to compute the power diagram, and its dual leads to the weighted Delaunay triangulation directly.

Given a set of points, the Voronoi cell $W_i$ at the point $q_i$ satisfies the following condition:

$$W_i = \{x | |x - q_i|^2 \leq |x - q_j|^2, \forall j\}.$$

If we associate every point $q_i$ with a power weight $w_i$, then the cell $W_i$ becomes the power Voronoi cell and satisfies the following inequality:

$$W_i = \{x | |x - q_i|^2 + w_i \leq |x - q_j|^2 + w_j, \forall j\}.$$

If the power weights of all points are the same, the power diagram is reduced to the normal Voronoi diagram, and the weighted Delaunay triangulation is reduced to the Delaunay triangulation. Figure 7a shows a Voronoi diagram, and Figure 7b exhibits a power diagram.
and its dual: the weighted Delaunay triangulation. In Figure 7b, the radii of the red circles illustrate the power weights of the points. We use the CGAL library [The CGAL Project 2015] to compute the power diagram.

In semi-discrete OMT framework, according to the equation 1, the power Voronoi cell is a function of unknown variables of \( h_i \), and we rewrite it into the following format:

\[
|x - q_i|^2 - 2h_i - |q_i|^2 \leq |x - q_j|^2 - 2h_j - |q_j|^2.
\]

Then we establish the relationship between the power weight \( w_i \) and the height variable \( h_i \) as:

\[
w_i = -2h_i - |q_i|^2
\] (5)

Let \( M = \{V,E,F\} \) represents the 3D dense mesh with vertex \( V \), edges \( E \) and faces \( F \). Denote \( M_e = \{V_e,E_e,F_e\} \) as the conformal 2D mesh with vertex \( V_e \), edges \( E_e \) and faces \( F_e \).

**Figure 8: The semi-discrete OMT.**

To apply the semi-discrete OMT framework, we need setup some necessary items as the follows:

1. We assign the uniform measure \( \mu = 1 \) to the source domain which is a disk.

2. The target domain is discrete and consists of a set of points which exactly are the vertex \( V_e \) of the 2D mesh \( M_e \). Their positions are the same as the \( V_e \). Their measure is one third of the total area of its neighbor triangles in the 3D mesh \( M \). We denote the discrete target points as \( V_e = \{q_1,q_2,\ldots,q_k\} \) and their corresponding measures are \( \nu = \{\nu_1,\nu_2,\ldots,\nu_k\} \), as shown in Figure 8.

3. The initial height vector in semi-discrete OMT is set as follows:

\[
h_i = -\frac{1}{2}(q_i,q_i), \ i = 1,2,\ldots,k.
\] (6)

4. According to the equation 3, the gradient vector of the OMT energy is the difference of the target measure and the areas of the current power Voronoi cells, it is computed as the following:

\[
\nabla E(h) = (v_1 - W_1, v_2 - W_2, \ldots, v_k - W_k)
\] (7)

5. Denote the edges (the blue edges in Figure 7) of the power Voronoi cells as \( e_j \) and their dual edges (the black edges in Figure 7) in the weight Delaunay triangulation as \( \tilde{e}_j \), then the Hessian matrix \( H(h) \) of the energy has the formula:

\[
H_{ij}(h) = \begin{cases}
\frac{|v_i - v_j|}{\rho_{ij}} & i \neq j, W_i(h) \cap W_j(h) \cap \Omega \neq \emptyset \\
\sum_{k \neq i} H_{ik} & i = j \\
0 & \text{otherwise}
\end{cases}
\] (8)

6. The OMT energy can be optimized by Newton’s method:

\[
H(h) \delta h = \nabla E(h),
\] (9)

then the height vector is updated by:

\[
h \leftarrow h + \lambda \delta h,
\] (10)

where \( \lambda \) is a step length parameter.

The detail of the optimization algorithm is shown in Alg. 1. After the optimization procedure stops, the area of every power Voronoi cell of the point \( q_i \) is exactly equal to its the target measure, i.e. one third of the total area of the adjacent triangles of vertex \( i \). Therefore the final optimal dual weighted Delaunay triangulation is locally area-preserving.

**Algorithm 1: Semi-discrete Optimal Mass Transportation Map**

**Input:** A set of discrete points \( Y = \{q_1,\ldots,q_k\} \), discrete target measure \( \nu = \{\nu_1,\ldots,\nu_k\} \).

**Output:** A partition of \( \Omega = \cup \Omega_i \), such that \( W_i \leftrightarrow q_i \) is the optimal mass transportation map.

1. Translate and scale \( Y \), such that \( Y \subset \Omega \).
2. Initialize the height vector \( h \) by the Eqn. 6.
3. Compute the power weights by the Eqn. 5.
4. Construct the power diagram \( V(h) \) of \( \Omega \).
5. Construct its dual weighted Delaunay triangulation \( \mathcal{T}(h) \).
6. for \( i \leftarrow 1 \) to \( k \) do
   - Compute the area of power Voronoi cell \( W_i(h) \).
   - Compute the gradient vector by Eqn. 7.
   - Compute the dual weighted Delaunay triangulation \( \mathcal{T}(h) \).
   - Compute the power diagram \( V(h + \lambda \delta h) \).
7. while \( \exists W_i(h + \lambda \delta h) \) is empty do
   - Adjust the step parameter by \( \lambda \leftarrow \frac{\lambda}{2} \).
   - Compute the power weights by the Eqn. 5.
   - Compute the power Voronoi diagram \( V(h + \lambda \delta h) \).
8. Calculate all edge lengths of \( V(h) \) and \( \mathcal{T}(h) \).
9. Build the Hessian matrix by Eqn. 8.
10. Solve the linear equation by Eqn. 9.
11. Compute the power diagram \( V(h + \lambda \delta h) \).
12. while \( V(W_i(h + \lambda \delta h)) \) is empty do
   - Break.

**return the mapping \( \{W_i(h) \leftrightarrow q_i, i = 1,2,\ldots,k\} \).**

In summary, the semi-discrete OMT based algorithm provides a tool to find a special partition of a given space with minimal cost, such that the area of every cell, i.e., the neighborhood area of the vertex in its dual triangulation, is equal to the specific target value. By this property, we can achieve a 2D mesh, where the neighborhood area of every vertex is equal to the corresponding area value in original 3D mesh. We show some conformal parameterizations in Figures 11e and 12e and area-preserving results adjusted by our semi-discrete OMT algorithm in Figures 11f and 12f. More results are demonstrated in Table 1.

6 Texture and Simplified Mesh Generation

After we have a local area-preserving parameterization that maps a dense mesh onto a 2D disk, we use two samplings to generate the normal map texture and the simplified mesh respectively. The first sampling is a high resolution one, which leads to a normal direction sampling in the dense mesh. The second one is low resolution, which samples a few of points in the dense mesh to create a simplified mesh.
As the 2D mesh is area-preserving, the number of sampling points in the neighborhood of every vertex is equal to its corresponding 3D counterpart, therefore, the normal directions of the 3D dense mesh will be uniformly sampled by our method.

Figure 10: (a) The sampling in the original 3D fish mesh; (b) the sampling in the 2D mesh; (c) the normal map texture

We sample the disk (or rectangle) with a high resolution grid, as shown in Figure 10b. The number of the grid points should be equal to or less than the number of the vertex of the dense mesh to capture its geometry details. Every grid point falls in a certain triangle. We compute it barycenter coordinates according to its containing 2D triangles. Then we map the sample point onto the corresponding 3D triangle, and compute its location by the same barycenter coordinates, as shown in Figure 10a. From it 3D location, we can obtain the normal direction of the sample point by interpolating the normal directions of three vertices of the 3D triangle. Finally we convert all normal direction values to RGB values and store them into a texture picture, as shown in Figure 10c and Figure 9d.

The simplification step also starts from sampling the same rectangle, but with a low resolution grid. The coordinates of the grid points will be the texture coordinates of the simplified mesh directly. As we sample the same rectangle as the one used in the texture creation step, these texture coordinates and the texture picture are mapped correspondingly. In contrast to the traditional method which generates a simplified mesh, then maps it to a 2D plane to obtain its texture coordinates, our method achieves the simplified mesh and its texture coordinates by one single step at the same time.

To obtain a better simplification mesh, firstly we sample the 2D OMT disk and compute the barycenter coordinates. Secondly we map the grid points from the OMT disk to the conformal disk. Although the locations of the grid points inside the OMT disk are uniform, they are non-uniform inside the conformal disk. Thirdly we triangulate these points into a mesh. Finally we map these points to the 3D dense mesh with the same barycenter coordinates. In Figure 4e and 4f, we show the dense and simplified meshes respectively.

7 Experiment

In this section, we demonstrate the efficiency and efficacy of our method. All the experiments were carried out on a laptop computer of Intel Core i5-4200 CPU, 2.29 GHz with 8GB memory. Our algorithm is implemented using C++ with visual studio 2015 on windows 10 OS. The power diagram is computed by [The CGAL Project 2015]. Our algorithm basically solves an optimization problem with Newton's method, and is simple to implement. In Figures 11 and 12, we demonstrate our normal map rendering results with two models. We render the simplified models (Figures 11b and 12b) with our normal map textures (Figures 11d and 12d), the results (Figures 11c and 12c) look similar with the high resolution ones (Figures 11a and 12a). More demos are exhibited in the table 1.

In Figure 9, we demonstrate the normal map rendering results of the simplified meshes with different resolutions. We observe that when the number of vertex of the simplified mesh is bigger than 1k, the normal map rendering results can not be improved more apparently.
In this paper, we apply the semi-discrete OMT algorithm in the application of the normal map rendering. Base on the area-preserving property of OMT parameterization, our method can sample the 3D high resolution meshes uniformly, therefore, result in a better normal map texture.

We also propose a different work flow to generate simplified meshes, their texture coordinates and normal map texture. For the different simplified meshes of the same dense model, our method only need to produce a single normal map texture.

Our method is practical, robust, and simple to implement. It can improve the performance of the current normal map method with a better rendering result. In the future, we will follow this direction to apply the semi-discrete OMT in exploiting the uniform sampling in mesh reconstruction or point based rendering.

When we minimize the energy, the power Voronoi cells changes continuously, however weighted Delaunay triangulation could have discrete jumping. It is possible to explore the continuous feature of the power Voronoi cells in other kinds of applications.

8 Conclusions

In Figure 13, we demonstrate the rendering result with a set of different curvature weights. The first row shows OMT disk with the different weights; the second row exhibits the corresponding simplified meshes; the third row displays their normal map rendering results. We observe that when the weights are bigger than 1.2, the simplification meshes will not change a lot, therefore the normal map rendering will look the same.

Acknowledgements

We would like to thank anonymous reviewers for their insightful feedbacks, valuable comments and suggestions. We also want to thank Yue Li for the help. Some pictures are rendered by Mitsuba [Jakob 2010]. The 3D models are the courtesy of AIM@SHAPE shape repository.

References

ALEXANDROV, A. D. 2005. Convex polyhedra. Springer Monographs in Mathematics. Springer-Verlag, Berlin.

AURENHAMMER, F. 1987. Power diagrams: properties, algorithms and applications. SIAM Journal on Computing 16, 1, 78–96.

BONNEEL, N., VAN DE PANNE, M., PARIS, S., AND HEIDRICH, W. 2011. Displacement interpolation using lagrangian mass transport. In ACM Transactions on Graphics (TOG), vol. 30, ACM, 158.

BONNEEL, N., PEYRÉ, G., CUTURI, M., MAZENC, F., MALISOFF, M., FRIDMAN, E., BONNET, C., MAZENC, F., DJEMA, W., LE HA VY NGUYEN, C. B., ET AL. 2016. Wasserstein barycentric coordinates: Histogram regression using optimal transport. ACM Transactions on Graphics 35, 4.

BONNOTTE, N. 2013. From knothe’s rearrangement to breiner’s optimal transport map. SIAM Journal on Mathematical Analysis 45, 1, 64–87.

BRENIER, Y. 1991. Polar factorization and monotone rearrangement of vector-valued functions. Communications on Pure and Applied Mathematics 44, 4, 375–427.

BUDDINSKIY, M., LIU, B., TONG, Y., AND DESBRUN, M. 2016. Power coordinates: a geometric construction of barycentric coordinates on convex polytopes. ACM Transactions on Graphics (TOG) 35, 6, 241.

BUSARYEV, O., DEY, T. K., WANG, H., AND REN, Z. 2012. Animating bubble interactions in a liquid foam. ACM Transactions on Graphics (TOG) 31, 4, 63.

CIGNONI, P., MONTANI, R., ROCCHINI, C., AND SCOPIGNO, R. 1998. A general method for preserving attribute values on
simplified meshes. In Visualization’98. Proceedings, IEEE, 59–66.

COHEN, J., OLANO, M., AND MANOCHA, D. 1998. Appearance-preserving simplification. In Proceedings of the 25th annual conference on Computer graphics and interactive techniques, ACM, 115–122.

DE BERGE, M., CHEONG, O., VAN KREVELD, M., AND OVERMARS, M. 2008. Computational Geometry: Algorithm and Application, 3rd ed. Springer-Verlag.

DE GOES, F., COHEN-STEINER, D., ALLIEZ, P., AND DESBRUN, M. 2011. An optimal transport approach to robust reconstruction and simplification of 2d shapes. In Computer Graphics Forum, vol. 30, Wiley Online Library, 1593–1602.

DE GOES, F., BREEDEN, K., OSTROMOUKHOV, V., AND DESBRUN, M. 2012. Blue noise through optimal transport. ACM Transactions on Graphics (TOG) 31, 6, 171.

DE GOES, F., ALLIEZ, P., OWHADI, H., AND DESBRUN, M. 2013. On the equilibrium of simplicial masonry structures. ACM Transactions on Graphics (TOG) 32, 4, 93.

DE GOES, F., WALLÉZ, C., HUANG, J., PAVLOV, D., AND DESBRUN, M. 2015. Power particles: an incompressible fluid solver based on power diagrams. ACM Transactions on Graphics (TOG) 34, 4, 50.

DESBRUN, M., MEYER, M., AND ALLIEZ, P. 2002. Intrinsic parameterizations of surface meshes. In Computer Graphics Forum, vol. 21, Wiley Online Library, 209–218.

FLOATER, M. 2003. One-to-one piecewise linear mappings over triangulations. Mathematics of Computation 72, 242, 685–696.

FU, X.-M., AND LIU, Y. 2016. Computing inversion-free mappings by simplex assembly. ACM Transactions on Graphics
Fu, X.-M., Liu, Y., and Guo, B. 2015. Computing locally injective mappings by advanced mips. ACM Transactions on Graphics (TOG) 34, 4, 71.

Garland, M., and Heckbert, P. S. 1997. Surface simplification using quadric error metrics. In Proceedings of the 24th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., 209–216.

Garland, M., and Shaffer, E. 2002. A multiphase approach to efficient surface simplification. In Proceedings of the conference on Visualization 02, IEEE Computer Society, 117–124.

Goes, F. D., Memari, P., Mullen, P., and Desbrun, M. 2014. Weighted triangulations for geometry processing. ACM Transactions on Graphics (TOG) 33, 3, 28.

Gortler, S. J., Gotsman, C., and Thurston, D. 2006. Discrete one-forms on meshes and applications to 3d mesh parameterization. Computer Aided Geometric Design 23, 2, 83–112.

Gu, X., and Yau, S.-T. 2003. Global conformal surface parameterization. In Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing, Eurographics Association, 127–137.

Gu, X., Gortler, S. J., and Hoppe, H. 2002. Geometry images. ACM Transactions on Graphics (TOG) 21, 3, 355–361.

Gu, X., Luo, F., Sun, J., and Yau, S.-T. 2016. Variational principles for minkowski type problems, discrete optimal transport, and discrete monge-ampere equations. Asian Journal of Mathematics (AJM) 2, 20 (Aprt.), 383–398.

Guskov, I., Vidimcë, K., Sweldens, W., and Schröder, P. 2000. Normal meshes. In Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., 95–102.

Haker, S., Zhu, L., Tannenbaum, A., and Angenent, S. 2004. Optimal mass transport for registration and warping. International Journal of computer vision 60, 3, 225–240.

Heckbert, P. S., and Garland, M. 1999. Optimal triangulation and quadric-based surface simplification. Computational Geometry 14, 1, 49–65.

Hoppe, H. 1999. New quadric metric for simplifying meshes with appearance attributes. In Proceedings of the conference on Visualization ’99: celebrating ten years, IEEE Computer Society Press, 59–66.

Hormann, K., and Greiner, G. 2000. Mips: An efficient global parametrization method. Tech. rep., DTIC Document.

Jakob, W. 2010. Mitsuba renderer. http://www.mitsuba-renderer.org.

Kantorovich, L. 1948. On a problem of monge. Uspekhi Mat. Nauk. 3, 225–226.

Kitagawa, J., Mérigot, Q., and Thibert, B. 2016. A newton algorithm for semi-discrete optimal transport. arXiv preprint arXiv:1603.05579.

Kovalsky, S. Z., Galun, M., and Lipman, Y. 2016. Accelerated quadratic proxy for geometric optimization. ACM Transactions on Graphics (TOG) 35, 4, 134.

Krishnamurthy, V., and Levoy, M. 1996. Fitting smooth surfaces to dense polygon meshes. In Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, ACM, 313–324.

Lévy, B. 2015. A numerical algorithm for 12 semi-discrete optimal transport in 3d. ESAIM M2AN (Mathematical Modeling and Numerical Analysis).

Liu, L., Zhang, L., Xu, Y., Gotsman, C., and Gortler, S. J. 2008. A local/global approach to mesh parameterization. In Computer Graphics Forum, vol. 27, Wiley Online Library, 1495–1504.

Liu, Y., Pan, H., Snyder, J., Wang, W., and Guo, B. 2013. Computing self-supporting surfaces by regular triangulation. ACM Transactions on Graphics (TOG) 32, 4, 92.

Melody, N. 2004. NVIDIA Melody. http://www.nvidia.com/object/melody_home.html.

Mérigot, Q. 2011. A multiscale approach to optimal transport. In Computer Graphics Forum, vol. 30, Wiley Online Library, 1583–1592.

Monge, G. 1781. Mémoire sur la théorie des déblais et des remblais. De l’Imprimerie Royale.

Rabin, J., and Papadakis, N. 2015. Convex color image segmentation with optimal transport distances. In International Conference on Scale Space and Variational Methods in Computer Vision, Springer, 256–269.

Rabinovich, M., Poranne, R., Panozzo, D., and Sorkine-Hornung, O. 2017. Scalable locally injective mappings. ACM Transactions on Graphics (TOG) 36, 2, 16.

Rubner, Y., Tomasi, C., and Guibas, L. J. 2000. The earth mover’s distance as a metric for image retrieval. International journal of computer vision 40, 2, 99–121.

Sander, P. V., Snyder, J., Gortler, S. J., and Hoppe, H. 2001. Texture mapping progressive meshes. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, ACM, 409–416.

Sander, P. V., Hoppe, H., Gortler, S., and Snyder, J. 2002. Signal-specialized parameterization.

Sheffer, A., Lévy, B., Mogilnitsky, M., and Bogoymakov, A. 2005. Abf++: fast and robust angle based flattening. ACM Transactions on Graphics (TOG) 24, 2, 311–330.

Sheffer, A., Praun, E., and Rose, K. 2006. Mesh parameterization and their applications. Foundations and Trends® in Computer Graphics and Vision 2, 2, 105–171.

Sheffer, A., Hormann, K., Lévy, B., Desbrun, M., Zhou, K., Praun, E., and Hoppe, H. 2007. Mesh parameterization: Theory and practice. ACM SIGGRAPH, course notes.

Smith, J., and Schaeffer, S. 2015. Bijective parameterization with free boundaries. ACM Transactions on Graphics (TOG) 34, 4, 70.

Snyder, J., Sander, P. V., Wood, Z. J., Gortler, S., and Hoppe, H. 2003. Multi-chart geometry images.

Solomon, J., Rustamov, R., Guibas, L., and Butscher, A. 2014. Earth mover’s distances on discrete surfaces. ACM Transactions on Graphics (TOG) 33, 4, 67.

Solomon, J., De Goes, F., Peyré, G., Cuturi, M., Butscher, A., Nguyen, A., Du, T., and Guibas, L. 2015.
Convolutional wasserstein distances: Efficient optimal trans-
portation on geometric domains. *ACM Transactions on Graphics (TOG)* 34, 4, 66.

Su, Z., Zeng, W., Shi, R., Wang, Y., Sun, J., and Gu, X. 2013. Area preserving brain mapping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2235–2242.

Su, Z., Wang, Y., Shi, R., Zeng, W., Sun, J., Luo, F., and Gu, X. 2015. Optimal mass transport for shape matching and comparison. *IEEE transactions on pattern analysis and machine intelligence* 37, 11, 2246–2259.

Su, K., Chen, W., Lei, N., Cui, L., Jiang, J., and Gu, X. 2016. Measure controllable volumetric mesh parameterization. *Computer-Aided Design*.

Su, K., Chen, W., Lei, N., Zhang, J., Qian, K., and Gu, X. 2016. Volume preserving mesh parameterization based on optimal mass transportation. *Computer-Aided Design*.

Su, K., Cui, L., Qian, K., Lei, N., Zhang, J., Zhang, M., and Gu, X. D. 2016. Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation. *Computer-Aided Geometric Design*.

Takeuchi, S., Kanai, T., Suzuki, H., Shimada, K., and Kimura, F. 2000. Subdivision surface fitting with qem-based mesh simplification and reconstruction of approximated b-spline surfaces. In *Computer Graphics and Applications, 2000. Proceedings. The Eighth Pacific Conference on*, IEEE, 202–212.

The CGAL Project. 2015. *CGAL User and Reference Manual*, 4.6.2 ed. CGAL Editorial Board. http://doc.cgal.org/latest/Surface_modeling/index.html.

Villani, C. 2008. *Optimal transport: old and new*, vol. 338. Springer Science & Business Media.

Weber, O., and Zorin, D. 2014. Locally injective parametriza-
tion with arbitrary fixed boundaries. *ACM Transactions on Graphics (TOG)* 33, 4, 75.

Xin, S.-Q., Lévy, B., Chen, Z., Chu, L., Yu, Y., Tu, C., and Wang, W. 2016. Centroidal power diagrams with capacity constraints: computation, applications, and extension. *ACM Transactions on Graphics (TOG)* 35, 6, 244.

Zhao, X., Su, Z., Gu, X., and Kaufman, A. 2013. Area-preserva-
tion mapping using optimal mass transport. *IEEE Transactions on Visualization and Computer Graphics* 19, 12, 2838 – 2847.
Table 1: Different shapes rendered using our normal map technique

| Original | Simplified | Normal Map | Texture | Conformal Disk | OMT Disk |
|----------|------------|------------|---------|----------------|----------|
| ![Original](image1) | ![Simplified](image2) | ![Normal Map](image3) | ![Texture](image4) | ![Conformal Disk](image5) | ![OMT Disk](image6) |
| ![Original](image7) | ![Simplified](image8) | ![Normal Map](image9) | ![Texture](image10) | ![Conformal Disk](image11) | ![OMT Disk](image12) |
| ![Original](image13) | ![Simplified](image14) | ![Normal Map](image15) | ![Texture](image16) | ![Conformal Disk](image17) | ![OMT Disk](image18) |
| ![Original](image19) | ![Simplified](image20) | ![Normal Map](image21) | ![Texture](image22) | ![Conformal Disk](image23) | ![OMT Disk](image24) |
| ![Original](image25) | ![Simplified](image26) | ![Normal Map](image27) | ![Texture](image28) | ![Conformal Disk](image29) | ![OMT Disk](image30) |