Computers & Graphics 35 (2011) 992–1000

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Technical Section

Meshless quadrangulation by global parameterization

Er Li [a], Bruno Lévy [b], Xiaopeng Zhang [a,]*, Wujun Che [a], Weiming Dong [a,b], Jean-Claude Paul [b,c]

[a] LIAMA-National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] INRIA-ALICE, France
[c] Tsinghua University, Beijing, China

ARTICLE INFO

Article history:
Received 13 October 2010
Received in revised form
20 May 2011
Accepted 20 May 2011
Available online 2 June 2011

Keywords:
Point cloud
Meshless parameterization
Quadrangulation
Surface reconstruction

ABSTRACT

Point cloud is a basic description of discrete shape information. Parameterization of unorganized points is important for shape analysis and shape reconstruction of natural objects. In this paper we present a new algorithm for global parameterization of an unorganized point cloud and its application to the meshing of the cloud. Our method is guided by principal directions so as to preserve the intrinsic geometric properties. After initial estimation of principal directions, we develop a kNN(*k*-nearest neighbor) graph-based method to get a smooth direction field. Then the point cloud is cut to be topologically equivalent to a disk. The global parameterization is computed and its gradients align well with the guided direction field. A mixed integer solver is used to guarantee a seamless parameterization across the cut lines. The resultant parameterization can be used to triangulate and quadrangulate the point cloud simultaneously in a fully automatic manner, where the shape of the data is of any genus.

1. Introduction

Point cloud has been widely used in CAD and computer graphics communities due to the availability of fast and accurate laser scan devices. Usually there is no topological information in the raw scan data. An amount of research work focuses on meshing the point cloud while keeping the original surface geometry. Although most of previous work can produce high quality triangular meshes, there is little consideration about how to control the shape and orientation of triangles in meshing the point cloud. On the other hand, more and more recent work focuses on how to convert an unstructured triangle mesh to a high quality quad mesh. Compared with a triangle mesh, a quad dominant mesh is more preferred due to its tensor-product nature desired in many applications, such as texturing, simulation with finite elements and B-spline fitting. And the quad meshes following principal directions are particularly appealing in modeling as they capture the symmetries of natural geometry. However, most of the methods are only applicable to meshes with explicit connectivity information. To get the quad mesh representation of a point cloud, a trivial way is first converting the point cloud to the triangular mesh with correct topology connection and then adopting the quadrangulation method designed for a triangle mesh. There are several problems about this straightforward approach:

- The conversion from a point cloud to a triangular mesh consumes computational resources; this problem is more critical when handling animation objects.
- Some geometry information may be lost during the process from a point cloud to a triangular mesh, such as the reconstruction method based on implicit form.
- A large polygon mesh also requires more space to store, so that this is not convenient to data transmission through network.

To overcome the above problems, we quadrangulate the point cloud directly through a meshless global parameterization. In this paper, we assume that the point cloud is uniformly distributed and well sampled, similar to the data obtained through laser scanners. Under this assumption, our key observation is that there is no need to convert a point cloud to a triangular mesh before the global parameterization. And then the resultant parameterization can be used to triangulate and quadrangulate the point cloud simultaneously in a fully automatic manner.

Our method is an extension of a global parameterization method [1] and we show that it is sufficient to quadrangulate a point cloud only utilizing its kNN graph which is widely used in geometric processing of point cloud. We design a smooth direction field across the point cloud and then calculate a global parameterization aligning well with the direction field. During those two stages of the algorithm we use only the connection information of kNN graph to measure smoothness of direction field on point cloud and compute the global parameterization. The advantage over the traditional way is that the resultant parameterization can be utilized to reconstruct a triangular mesh and a quad mesh at the same time and thus reduces the overall computation cost.

Our main contributions can be summarized as follows:

- A kNN graph-based algorithm to construct smooth direction field on point cloud with main geometric feature alignment;

---

* Corresponding author.
E-mail address: xpzhang@nlpr.ia.ac.cn (X. Zhang).

0097-8493/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cag.2011.05.003

- An extension of global parameterization method from a triangle mesh to a point cloud;
- Applications of the parameterization in triangle meshing and quad meshing.

2. Related work

Quad remeshing. Much work has been developed on quadrangulation of mesh models due to their wide advantages in various application fields. More details can be found in [2]. Here we only review the most related work.

Alliez et al. [3] remesh a triangular mesh into a quad dominant mesh by flattening it onto a 2D parametric plane via a discrete conformal parameterization and then by intersecting lines of curvature properly integrated in the plane. Marinov and Kobbelt [4] improve this method by directly integrating lines of curvature on the triangular model, so that it can be used on a surface of arbitrary genus. Dong et al. [5] build a harmonic scalar field, and the gradients of this field provide a smooth vector field for quadrilateral remeshing. Dong's method generates a more regular mesh but sacrifices feature alignment.

Another efficient way of quadrangulation is to decompose a mesh model with complex shape into several patches, and then to convert them into quadrilaterals, respectively. Boier-Martin et al. [6] propose a clustering-based method to decompose the surface. Dong et al. [7] construct the Morse–Smale complex of Laplacian eigenfunctions to form a quadrangular base mesh. The function distributes extrema evenly across a mesh surface and thus the final quadrilateral result is well shaped and with few singularities. However, the feature alignment is not guaranteed. Tong et al. [8] use a singularity graph to control the alignment and design the quadrangulation with discrete harmonic forms to create quads. Based on their work of [7], Huang et al. [9] propose a controllable spectral method to remesh a triangular mesh. By enforcing orientation and alignment constraints upon Laplacian eigenfunctions, better feature alignment is generated. Recently, the work in [10] is able to remesh a surface into anisotropically sized quads based on standing wave construction and quasi-dual MSC extraction [7].

Global parameterization has proven to be another useful tool for quadrangulation. In [11] a periodic global parameterization guided by principal directions is proposed to parameterize the input triangular model. Then the quadrilateral mesh can be obtained by tracking the iso-lines in the parameterized domain. This method generates a high quality quad dominant meshes automatically with little user interaction. In [12], they convert a given frame field into a single vector field on a branched covering space. Then the surface is cut and a global parameterization guided by the frame field is calculated to produce a quadrilateral mesh.

While all of the mentioned methods are performed on mesh data, the research is scarce yet on how to design a quadrangulation of point cloud. Kalogerakis et al. [13] extract lines of curvature from a noisy point cloud and these lines then can be used for the direct reconstruction of a quad dominant mesh. This method, similar to that in [4], inspires our method for direct quadrangulation of pure point cloud data.

Meshless parameterization. While much work has been done on mesh parameterization, there is only a little work focusing on meshless parameterization. Some basic methods were discussed in Floater and Reimers [14] to parameterize unorganized point sets. These methods yield good results on the point surfaces with disk topology. In [15], spherical parameterization is applied to mesh a 3D point cloud to a manifold genus-0 mesh. Tewari et al. [16] extend the work of Gu and Yau [17] to parameterize genus-1 point set surface using discrete one-forms. More generally, Guo et al. [18] realize global conformal parameterization on a point set surface and apply the parameterization on thin-shell simulation.

3. Background work and overview of algorithm

Our approach performs global parameterization directly on a raw and noisy point cloud, and then the resultant parameterization is used to construct a curvature-aligned quad mesh. Here we first give a brief introduction about the work [1] and then show the overview of our adaptations. The quadrangulation of a triangle mesh consists of two main steps: direction field construction and global parameterization. In the first stage, a set of salient directions are selected by measuring their relative anisotropy, and then a smooth direction field that take these directional constraints into account is solved in terms of a mixed integer problem. In the second stage, the mesh is cut into a disk-like surface with all singularities lying at the boundary. Subsequently two scalar fields whose gradients follow the direction field is computed as the final parameterization. Two compatibility conditions across boundary are incorporated into the parameterization as linear constraints to guarantee a seamless global parameterization: first, the mismatch between parameter values across the cut edge should be integer; second, the gradients of parameter values meet the rotations by multiples of $\pi/2$. Additionally, to make a pure quad mesh, the singularities are required to be at integer locations.

To provide a robust and flexible framework for quadrilateral meshing of a point set surface, our algorithm proceeds in three steps.

- First, we estimate the curvature tensor of each point to deduce two principal directions, and we smooth these direction fields globally to make the directions more consistent (Section 4).
- Second, we define an energy function and compute two scalar functions by minimizing this energy so as to make gradients of scalar functions best fit the principal directions (Section 5).
- Finally, a quadrilateral mesh is constructed according to the parameterization obtained in the second step (Section 6).

We will describe in detail how we do in the first and second steps on a point cloud, and the experiment results will be shown and analyzed in Section 7.

4. Direction field construction on point cloud

Principal directions serve as a good start to construct the direction field, but the question is that we can usually get high-quality estimation of principal directions in a curved surface region and the principal directions are not reliable in a flat one so that the corresponding estimation is less meaningful for our purpose. Our idea for this is defining the directions in these areas by smoothing the direction field globally. For the point cloud the difficulty is how to measure the smoothness of the direction field and identify singularities since the face-based method is not suitable here. In this section we address this problem by utilizing kNN graph.

Smoothness on kNN graph. For a point set surface, a variety of methods have been developed to calculate curvature tensor as accurate as possible. We only use the direction field to reflect the rough geometric features, so a common method is accurate enough. Here we adopt normal fitting method [19] to estimate the normal and curvature tensor on each point.

The smoothness of a direction field should reflect the continuous variations of directions on nearby points. So we employ the concept of parallel transport [20] in differential geometry on the kNN graph of the point cloud. Let $\mathbf{p}$ and $\mathbf{q}$ be two points on the point set surface, and let $\mathbf{V_p}$ and $\mathbf{V_q}$ be two vectors on $\mathbf{p}$ and $\mathbf{q}$, respectively, $\mathbf{T_p}$ and $\mathbf{T_q}$ be the projections of $\mathbf{p}-\mathbf{q}$ to the tangent plane at $\mathbf{p}$ and $\mathbf{q}$. Then $\mathbf{V_p}$ is said to be equivalent to $\mathbf{V_q}$ if the angle between $\mathbf{V_p}$ and $\mathbf{T_p}$ is equal to the angle between $\mathbf{V_q}$ and $\mathbf{T_q}$. According to this definition we measure the smoothness of the