J. Vis. Commun. Image R. 20 (2009) 190–203

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

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

Generating segmented meshes from textured color images

Mario A.S. Lizier [a,*], David C. Martins Jr. [b], Alex J. Cuadros-Vargas [a], Roberto M. Cesar Jr. [b], Luis G. Nonato [a]

[a] ICMC – Universidade de São Paulo, Av. Trab. São Carlense, 400, P.O. Box 668, 13560-970 São Carlos, SP, Brazil
[b] IME – Universidade de São Paulo, Rua do Matão, 1010 - Cidade Universitária, 05508-090 São Paulo, SP, Brazil

ARTICLE INFO

Article history:
Received 13 May 2008
Accepted 13 January 2009
Available online 31 January 2009

Keywords:
Mesh generation
Delaunay triangulation
Feature evaluation and selection
Texture classification
W-operators
Texture segmentation
Imesh image
Mesh modeling
Mesh generation from image data
Mesh segmentation

ABSTRACT

This paper presents a new framework for generating triangular meshes from textured color images. The proposed framework combines a texture classification technique, called W-operator, with *Imesh*, a method originally conceived to generate simplicial meshes from gray scale images. An extension of W-operators to handle textured color images is proposed, which employs a combination of RGB and HSV channels and Sequential Floating Forward Search guided by mean conditional entropy criterion to extract features from the training data. The W-operator is built into the local error estimation used by *Imesh* to choose the mesh vertices. Furthermore, the W-operator also enables to assign a label to the triangles during the mesh construction, thus allowing to obtain a segmented mesh at the end of the process. The presented results show that the combination of W-operators with *Imesh* gives rise to a texture classification-based triangle mesh generation framework that outperforms pixel based methods.

Crown Copyright © 2009 Published by Elsevier Inc. All rights reserved.

1. Introduction

The problem of generating meshes from images has been investigated in two different contexts, namely, image representation and numerical simulation. The techniques devoted to image representation (also called image mesh modeling) aim at modeling a given image as a mesh, envisioning mainly a more compact representation instead of the pixel representation. Image meshing methods toward numerical simulation, however, intend to generate meshes suitable for numerical solution of physical phenomena, as for example, blood flow simulation and mechanical simulation of lung behavior.

Although targeting distinct applications, both approaches make use of a similar set of conventional image processing tools, such high-pass filters combined with thresholding, iso-values and interpolation in order to decide where the vertices of the mesh must be placed. As expected, the use of conventional tools limits the effectiveness of image based mesh generation methods to the restrictive class of non-textured images. In fact, most techniques strongly rely on user intervention when dealing with textured images. This is the case of most techniques in the context of numerical simulation, as many mesh generation techniques assume well defined curves representing the boundary of the domain to be meshed as input, what is very difficult to be obtained through conventional segmentation methods.

In order to circumvent these difficulties, this work proposes a new methodology for meshing textured color images that reduces the user intervention drastically, being thus a valuable alternative to the existing methods. In fact, by exploiting the concept of W-operators [1] and the developed image based mesh generation technique called *Imesh* [2], a new framework for handling textured color images has been conceived. Besides dealing with color and texture, our approach offers a very flexible mesh generation mechanism which can be employed to image mesh modeling as well as to image mesh generation for numerical simulation. W-operators are employed both for vertex placement and cell labeling, thus resulting in an unified scheme to generate a segmented mesh. A quality mesh is then obtained by applying an adaptation of Ruppert's algorithm [3] to the segmented mesh. Therefore, the framework proposed in this work produces a segmented quality mesh representing structures contained in textured color images, a feature not found in other image based mesh generation techniques.

The approach proposed in this paper is based on a supervised texture classification, called W-operator [1], and it is oriented by a Delaunay mesh created by *Imesh* [2] algorithm. The W-operator-based approach proposed in the present paper is intrinsically local since the image is seen locally through a window that just observes a limited part of the image, allowing to take into account the texture information. In fact, our supervised approach allows a straightforward partitioning of the resulting mesh without requiring an user defined parameter to specify the number of partitioned regions [2,4]. Furthermore, the robustness and effectiveness sup-

* Corresponding author.
E-mail address: lizier@icmc.usp.br (M.A.S. Lizier).

1047-3203/$ - see front matter Crown Copyright © 2009 Published by Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2009.01.002

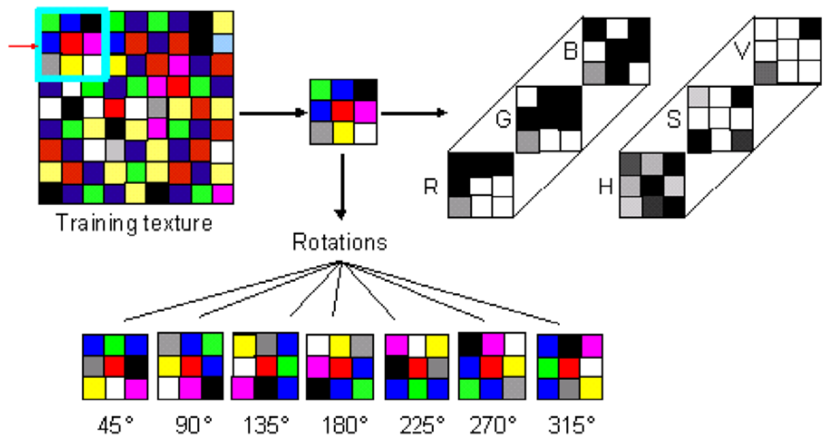

Fig. 1. Generation and replication of the training samples by rotations of 45° ($d = 45$).

plied by the Delaunay triangulation jointly with an effective vertex insertion mechanism that avoid bad quality triangles render *Imesh* a very reliable mesh generation scheme. Moreover, *Imesh* makes use of a theoretically guaranteed mesh refinement scheme [5] a characteristic not found in other methods devoted to generate meshes from images.

Before presenting the main aspects of our approach, we briefly describe the related work in the next section. The *Imesh* technique, W-operators and their combination are presented in Section 3. Section 4 describes the methodologies employed to segment and to improve the mesh. Results and conclusions are discussed in Sections 5 and 6, respectively.

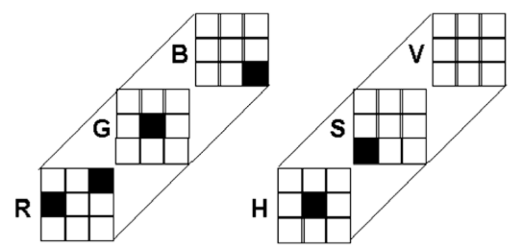

Fig. 2. An example of a sub-window from which the W-operator may be designed. The chosen sub-window is composed by the black positions.

2. Related work

In this section we present a review of relevant methods for generating triangular meshes in domains defined by image. Our discussion is focused on two-dimensional algorithms, although some of the presented techniques can be extended to handle 3D images. The related techniques are divided in two main groups, as following.

Image Mesh Modeling techniques intend to build a mesh that minimizes an error measurement, usually the approximation error between the original image and that represented by the triangular mesh. Garcia et al. [6], for example, have presented an algorithm that controls the maximum root-mean-square error (RMS) by choosing the vertices of the mesh from a curvature image, i.e., more vertices are placed in areas with high curvatures. The mesh model is built by generating the Delaunay triangulation [7] from the chosen vertices. Regions with high RMS error are re-sampled and new vertices are added to the Delaunay triangulation. Garcia's method is a typical example of an adaptive approach, which is characterized by beginning with an initial mesh that is iteratively refined in order to reduce the interpolation error. Many algorithms devoted to represent images by meshes are based on adaptive approaches [8–11]. Alternatively, some techniques have adopted an opposite strategy, i.e., a fine mesh is successively coarsened until the approximation error reaches a tolerance [12]. Mixed approaches that combine refinement and coarsening [13,14] as well as optimization schemes for positioning the vertices have also been developed [15,4].

Still envisioning image mesh modeling, Yang et al. [16] proposed an one pass method that makes use of zero-crossings jointly with error diffusion in order to choose and to position a set of vertices from which the Delaunay triangulation is built. Besides reducing the approximation error, the authors argue that this strategy produces meshes of good quality.

A main drawback of the above methods in the context of mesh generation for textured images is the difficulty to define an appropriated approximation error which does not generate an excessive number of elements. This effect happens when internal details of the texture are detected and new elements are created to represent them.

In general, *Image Modeling for Simulation* methods divide the mesh generation process in two main steps: pre-processing and mesh generation. The pre-processing step aims at filtering and segmenting the image in order to detect the regions of interest, which are "meshed" in the mesh generation step. Cebral and Lohner [17] binarize the original image in order to extract well defined contours from which the mesh is built. Binarization has also been employed as a pre-processing strategy by Zhang et al. [18] and Berti [19]. In both algorithms the mesh is generated by defining an implicit function from the binary images that guides a space partitioning strategy (quadtree) and thus the mesh generation. They also add a post-processing step to improve the quality of the mesh.

By making use of pre-processing to reduce noise and to highlight sharp features, Hale [20] proposed the use of a potential energy function to align a lattice of points with the image features. The mesh is then generated by Delaunay triangulation from the aligned points. The main problem with Hale's strategy is that distinct regions are not identified. Therefore, a mesh segmentation post-processing step is required in order to distinguish the different structures contained in the image.

To the best of our knowledge, the work by Hermes and Buhmann [4] is one of the few examples of mesh generation technique that can to deal with textured images. The triangulation scheme proposed in [4] is based on a minimization mechanism that relies on an iterative refinement approach and vertices's movement. Different from our approach, the quality of generated elements is not considered and the resulting mesh is not Delaunay. Moreover, the cost function to be minimized is defined without any previous

Download English Version:

https://daneshyari.com/en/article/529435

Download Persian Version:

https://daneshyari.com/article/529435

Daneshyari.com