Graphical Models 82 (2015) 123–136

Contents lists available at ScienceDirect

Graphical Models

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

Gaussian curvature using fundamental forms for binary voxel data[☆]

Markus Kronenberger[a], Oliver Wirjadi[b,*], Johannes Freitag[c], Hans Hagen[a]

[a] Computer Science Department, University of Kaiserslautern, Germany
[b] Fraunhofer ITWM, Kaiserslautern, Germany
[c] Alfred-Wegener-Institut, Bremerhaven, Germany

ARTICLE INFO

Article history:
Received 25 October 2014
Revised 15 June 2015
Accepted 16 June 2015
Available online 25 June 2015

Keywords:
Image analysis
Computed tomography
Geometry processing
Ice
Gaussian curvature
Fundamental forms
Segmentation

ABSTRACT

Local curvature characterizes every point of a surface and measures its deviation from a plane, locally. One application of local curvature measures within the field of image and geometry processing is object segmentation. Here, we present and evaluate a novel algorithm based on the fundamental forms to calculate the curvature on surfaces of objects discretized with respect to a regular three-dimensional grid. Thus, our new algorithm is applicable to voxel data, which are created e.g. from computed tomography (CT). Existing algorithms for binary data used the Gauss map, rather than fundamental forms. For the calculation of the fundamental forms, derivatives of a surface in tangent directions in every point of the surface have to be computed. Since the surfaces exist on grids with restricted resolution, these derivatives have to be discretized. In the presented method, this is realized by projecting the tangent plane onto the discrete object surface. The most important parameter of the proposed algorithm is the size of the chosen window for the calculation of the gradient. The size of this window has to be selected according to object size as well as with respect to distances between objects. In our experiments, an algorithm based on the Gauss map provided inconsistent values for simple test objects, whereas our method provides consistent values. We report quantitative results on various test geometries, compare our method to two algorithms working on gray value data and demonstrate the practical applicability of our novel algorithm to CT-reconstructions of Greenlandic firn.

1. Introduction and related work

Curvature is one of the fundamental properties of any geometric object. While different definitions exist, the term curvature in three dimensional space always describes the degree to which a surface deviates from a plane. For voxel data, algorithms for the computation of integral curvature measures, i.e., measures which characterize the entity of an object's surface, are long established and highly efficient, see e.g. [1]. In fact, the integrals of mean and Gaussian curvature are two of the four intrinsic volumes of any three dimensional non-empty convex set, and can be extended to a very general class of objects [2, chap. 14].

Yet, such integral measures do not suffice for applications such as object segmentation [3–5] or recognition [6], where local curvature measurements are required. For surface meshes, i.e., polygon meshes describing the shape of 3D-objects, the concept of local curvature is a widely studied subject, see e.g. [7–10]. These algorithms use different approaches: Hamann used local bivariate quadratic approximations over the tangent plane [7], while Theisel et al. compute curvature from the three normal vectors given on the nodes of each triangular surface element [8].

☆ This work was partially funded by the German Federal Ministry of Education and Research through project 05M2013 (AniS).

* Corresponding author.
E-mail address: oliver.wirjadi@itwm.fraunhofer.de (O. Wirjadi).

http://dx.doi.org/10.1016/j.gmod.2015.06.009
1524-0703/

The present paper concerns local curvature in binary voxel data produced e.g. by computed tomography, electron tomography, magnetic resonance imaging or serial sectioning techniques. We do not consider the binarization step in the present paper, but will rather assume that the data is given in a 0/1-representation, where voxels with value 0 will mark the background locations. Such situations occur frequently. For example, in materials science applications, the volume fractions of the material constituents are sometimes known a priori. Thus, datasets can be preprocessed to represent the correct volume fraction of the material. Then, one possible way to compute local curvature would be to construct surface meshes via algorithms such as marching cubes [11] or the wrapper algorithm [12] and to apply one of the aforementioned algorithms. But this route has major obstacles: Surface meshes generated from voxel data are often large and subsequent mesh simplification is necessary (e.g. [13]). Such a simplification reduces the amount of data, but can lead to a loss of structural details. Furthermore, local curvature measures on the surfaces in such binary data enable curvature-based segmentation of pores or particles, cf. Section 5.

Thus, it would be desirable to compute local curvature directly on voxel grids. This problem has already been studied by other authors, especially in the context of segmenting porous networks in snow and ice—the same application domain as in the present paper. Brzoska, Flin and their coauthors first proposed to fit circles to surfaces using 3D-distance maps created using a Chamfer-metric based distance transformation [14]. There, they were able to compute mean curvature, only, and results were noisy even on smooth surfaces. Therefore, they studied Gaussian curvature in voxel data in a more recent series of publications [3,4,15]. In all of these latter publications, they use the Gauss map to compute Gaussian curvature. This would require partial derivatives in off-grid directions, which the authors of [3,4] described as computationally too expensive. In order to make the required computation of gradients more efficient, their algorithm rotates every local tangent system of the surfaces under study onto the image's coordinate axes by selecting the coordinate direction which is closest to the normal direction in every surface point. Their approximative method was not accurate enough for broad applicability, in our experiments, cf. Section 4.

Furthermore, there exist a number of highly useful methods to compute curvature measures on surfaces implicitly defined by the iso-contours of three-dimensional gray value data. One interesting approach with very good results has been given by Thirion and Gourdon [16]. But in their approach, they need to consider special cases due to their local parameterization of the surface in the form $f(u, v)$. Thus, they have to symmetrize their equations [16]. This requires them to consider at least one so-called "privileged direction", where the gradient has to be co-linear to this privileged direction. In our general approach, we do not have to consider any special cases of this kind. Another approach, which unlike ours and Thiron and Gourdon's does not use fundamental forms, has been proposed by Rieger et al. [17]. Instead, their method is based on the use of the gradient structure tensor and Knutsson's tensor representation of orientation [18]. Due to the latter, Rieger et al.'s approach requires special post-processing to retrieve the signs of the principle curvatures. This can be achieved either via the Hessian matrix as in the original paper [17], or using alternative approaches [19]. Our method does not require any such post-processing. An experimental comparison to our method can be found in Section 4. More recently, Coeurjolly, Lachaud and Levallois have proposed a curvature estimator based on integral invariants [20]. They were able to proof multi-grid convergence for their method and show results comparable to previous methods on some test surfaces.

In the following, we propose a novel and accurate algorithm for computing fundamental forms – and therewith Gaussian curvature – for every point on the surface of objects discretized on a regular grid. In differential geometry, the fundamental forms [21] are used to describe intrinsic and extrinsic properties of surfaces. The first fundamental form allows e.g. to compute the length of curves and the area of a region on a surface while adding the second fundamental form allows to compute extrinsic surface properties such as principal curvatures, as well. This paper presents an extended version of the first author's bachelor thesis [22].

First, we introduce our method to compute the first and second fundamental forms on a voxel grid in Section 2. We evaluate it on various test objects in Section 4. Subsequently, we apply our algorithm within the curvature-based segmentation framework from [3,4] in Section 5 to the analysis of real structures measured by X-ray computed tomography in Section 6.

2. Discrete fundamental forms

All algorithms described in the present paper have been developed for discrete three dimensional data. That is, for data given in form of a binary input image $f : \mathbb{Z}^3 \to \{0, 1\}$, where 0 indicates background, whereas the surfaces of interest are given by the boundary of the foreground—which corresponds to the set of voxels with value 1. Note that all results can be transferred to label images with values in $\mathbb{N}$ in a straightforward manner.

But first, we will review the theoretical background in a continuous setting. For a comprehensive introduction to the field of differential geometry, we refer to the monographs by do Carmo [21] and by Kreyszig [23]. We consider a regular surface S (see e.g. [21, p. 52]) in three dimensional Euclidean space defined by a mapping $X : \mathbb{R}^2 \to \mathbb{R}^3$. Thus, X maps any point $(u, v) \in S$ to a point $X(u, v) \in \mathbb{R}^3$. By $p = (u, v) \in S$ we will denote some point on that surface which corresponds to a point $(x, y, z) = X(u, v) \in \mathbb{R}^3$. Furthermore, we will write X_u and X_v for the partial derivatives of the surface in directions u and v, respectively, and we will use notations such as X_{uv} for second partial derivatives, analogously.

2.1. First fundamental form

The first fundamental form enables local measurements on a surface without referring to the surrounding space $\mathbb{R}^3$. For example, the length of a curve on a surface, the angle between two curves on a surface and the surface area of a region on a surface can be determined.

Definition 1 (first fundamental form [21, p. 92]). Let S be a regular surface. Thus, a tangent plane $T_p(S)$ exists in every