Computers & Graphics 52 (2015) 116–128

Contents lists available at ScienceDirect

Computers & Graphics

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

Special Section on SIBGRAPI 2014

Sketch-based modeling and adaptive meshes

Emilio Vital Brazil [a,*], Ronan Amorim [a], Mario Costa Sousa [a], Luiz Velho [b], Luiz Henrique de Figueiredo [b]

[a] *Department of Computer Science, University of Calgary, Canada*
[b] *IMPA – Instituto Nacional de Matemática Pura e Aplicada, Rio de Janeiro, Brazil*

ARTICLE INFO

Article history:
Received 28 November 2014
Received in revised form
6 June 2015
Accepted 17 July 2015
Available online 14 August 2015

Keywords:
Sketch-based modeling
Adaptive mesh
Geometric modeling

ABSTRACT

We present a theoretical approach to the problem of sketch-based surface modeling (SBSM) and introduce a framework for SBSM systems based on adaptive meshes. The main advantage of this approach is a clear separation between the modeling operators and the final representation, thus enabling the creation of SBSM systems suited to specific domains with different demands. To support the proposed approach we present two different sketch-based modeling systems built using this framework. The first one has the capability to control local and global changes to the model; the second one follows geological data constraints. To build the first system that provides the user with control of local modifications we developed a mathematical theory of vertex label and atlas structure for adaptive meshes based on stellar operators.

1. Introduction

Sketches are the most direct way to communicate shapes: humans are able to associate complex shapes with few curves. However, sketches do not have complete shape information, and the information sketches do provide is often inexact; thus, ambiguities are natural. On the other hand, to create, edit, and visualize shapes using computers, we need precise mathematical information, such as a function formula or a triangle mesh. The problem of how to model shapes using sketches can be formulated as how to fill the missing information about the model. In the last 15 years, sketch-based modeling (SBM) has become a well established research area, encompassing work in different domains, such as computer vision, human–computer interaction, and artificial intelligence [1]. However, this body of work lacks a more theoretical approach on how to build a sketch-based modeling system for a given application. In contrast, in this work we introduce a framework tailored for sketch-based surface modeling (SBSM) taking advantage of adaptive meshes. Based on the proposed framework we present and discuss two different sketch-based modeling systems.

We advocate that SBSM systems must be suited to each specific application: the specificities of a certain field require suitable mathematical representations for the domain model, and this plays a central role in the characterization of SBSM applications. However, there are common requirements in many SBSM applications that can be abstracted to guide the definition of specific representations for specific domains. These requirements have three main aspects: (1) *dynamic* – the surface will change during the modeling process; (2) *interactive* – the user must be able to see the model changing with interactive response and feedback; (3) *controlled freedom* – some applications have specific modeling rules and the systems must be able to incorporate these rules to guide the user in building a correct model, without losing flexibility.

Adaptive meshes are generally associated with the ability to produce detailed complex models using a smaller mesh. However, our proposed framework is based on adaptive meshes because they can be dynamic and enable rapid updates with local control. Different schemes of adaptive meshes can be used to create a system using our framework; indeed, the choice of the scheme must take into account the final application requirements, such as how to represent features, what changes of topology are allowed, and how smooth the models need to be. Fig. 1 shows an instance of a model built within our framework: a 4–8 adaptive mesh adapted to an implicit surface.

Based on the proposed framework we built two majorly different sketch-based modeling systems which are presented in this paper. The two sketch-based modeling systems that will be presented here are built using our proposed framework and have major differences. The first system is the *Detail Aware Sketch-Based Surface Modeling* (DASS, Section 5), which approaches a common problem in many SBSM systems: the lack of good control of global and local transformations. We created DASS to allow us to validate

---

* Corresponding author.
E-mail address: evbrazil@ucalgary.ca (E. Vital Brazil).

http://dx.doi.org/10.1016/j.cag.2015.07.020
0097-8493/

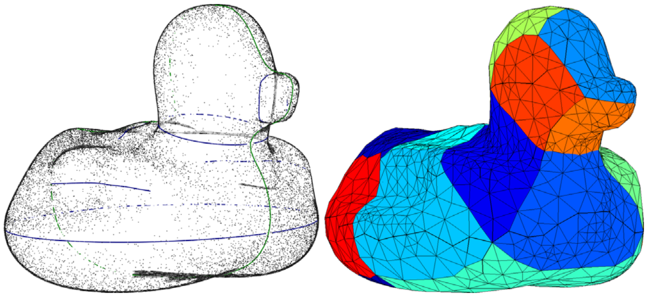

Fig. 1. A rubber duck modeled using DASS system: the Hermite Radial Basis Function (HRBF) implicit surface (left) and the adapted 4–8 mesh (right).

our proposed framework, exploring the limitations of a general system without a well defined task. To achieve the required control we developed a method to create atlas structures for adaptive meshes based on stellar operators [2]. The second system is the *Geological Layer Modeler* (Section 6), which is a sketch-based system specialized for geology that aims to help geophysicists to create subsurface models. This system is a good illustration of controlled freedom, where the sketch operators should be restricted to follow geological rules.

2. Related work

In the past decades there has been a large body of work in sketch-based surface modeling [3–7]. However, these systems are more concerned with the final results and do not consider the theoretical aspects of the mathematical surface representation used. We discuss below the main works on free-form sketch-based surface modeling that start from scratch under the light of its representations.

There are many ways to represent surfaces in $\mathbb{R}^3$. The most common and general are parametric representations and implicit representations. However, in order to be used in computer graphics and modeling applications, these representations must be more specific and possess practical qualities. As examples we can cite the BlobTree [8], piecewise algebraic surface patches [9], convolution surfaces [10], generalized cylinders, polygonal meshes, subdivision surfaces, among others.

Teddy [13], Fibermesh [4], and Kara and Shimada [11] use triangle meshes as a base representation for their modeling systems. Teddy and Fibermesh start with a planar curve and create an inflated mesh based on the curve's geometry. Teddy supports extrusion and cutting operators that cut a mesh part, then create a new mesh patch, which is merged with the model. Similarly, Fibermesh creates a new mesh based on the input sketches and places it using optimization on differential coordinates, thus enabling the system to keep all previous strokes as constraints. Kara and Shimada also keep a set of 3D curves to define the final model. However, they use curve loops to define triangle mesh patches that have minimum curvature, instead of optimizing across the whole mesh. These patches can be modified using physically based deformation tools. These three systems are based on the triangle mesh representation and use it to build their modeling operators; as result, their advantages and limitations are directly related with that chosen representation.

Using triangle meshes for modeling purposes has several advantages over other representations. First, triangle meshes are largely used by both academia and industry, and most graphics pipelines are based on triangles, which means that what you see is what you get. Moreover, there is much research on triangle meshes and many techniques have been developed for creating and editing meshes. On the other hand, applying these techniques in sketch-based modeling is not a straightforward task: techniques must be chosen based on the application scope, and these choices will define the limitations of the system. These limitations are noticeable in Teddy and Fibermesh – the latter approaches some drawbacks of the former using optimization on differential representation. Compared with Teddy, in Fibermesh the mesh quality is improved, the topology can be changed, and the construction curves are maintained using differential mesh techniques. However, the need for global optimization to assure mesh quality removes control over global and local editions: editing a small part of the model could affect other parts. Indeed, Nealen et al. [4] and Kara and Shimada [11] raised this issue: Nealen et al. suggested to embed the multi-resolution operator as a solution, whereas Kara and Shimada suggested to improve their method of creating and editing curves.

Parametric surfaces are defined by mapping a planar domain to 3D space. Working with parametric surfaces has some advantages: it is simple to obtain a good triangle mesh that approximates the model, it is relatively easy to map textures to the surface, and it provides continuous normal and curvature information. Cherlin et al. [12] and Gingold et al. [5] use parametric representation to create sketch-based systems. Cherlin et al. introduce two novel parametric surfaces based on sketched curves; Gingold et al. convert sketches to generalized cylinders. However, both approaches have issues with topology change and creating augmentations; these difficulties are mainly caused by the chosen parametric representations. Nasri et al. [13] and Orbay and Kara [7] create their systems based on subdivision surfaces – only being able to deal with set of curves that form closed loops. Heightfield is another example of parametric surface: it gives a 3D point (x, y, z) as a function of 2D coordinates, $z = f(x, y)$. This representation is fast and simple, and is usually enough for most terrains comprising mountains and hills. However, heightfields are not able to represent terrains with more complex geological structures, such as overhanging cliffs or caves. Hnaidi et al. [14] present a sketch-based system to model terrains. The characteristics of the terrain are defined by the user through a set of feature curves representing ridges, river beds, and cliffs. Constraints on these curves define elevation, angle and noise parameters along them. These constraints are then defined for the entire domain by diffusion. When the smooth terrain is ready, details are added by a procedural noise generator. The final terrain is a heightfield that results from combining the smooth terrain with the details.

In contrast with parametric surfaces, implicit surfaces can easily change topology when parameters change. They can also provide a compact, flexible, and mathematically precise representation which is well suited to describe coarse shapes. The implicit representation used for modeling surfaces has good modeling aspects providing important and mathematically precise information. Implicit surfaces allow global calculations, such as point classification (i.e., whether a point is inside or outside the surface volume) and distance evaluation. They also provide with access to local differential properties, such as normals and curvature. Karpenko et al. [15] introduced variational implicit surfaces as representation to sketch-based surface modeling. Vital Brazil et al. [6] improved this formulation by adding normals as hard constraints. Amorim et al. [16] presented a sketch-based system using Hermite–Birkhoff interpolation to create implicit models applied to geology. Araujo and Jorge [17] provided a set of sketch-based operators adapting the multi-level partition-of-unity implicit model [18]. Schmidt et al. [19] used BlobTrees as a main representation of the ShapeShop system. Bernhardt et al. [20] built the Matisse system based on convolution surfaces. These systems share the main disadvantages known about implicit representations: (1) the standard graphics pipeline is not prepared to handle implicit models; (2) few industrial processes use implicit surfaces,