A MULTIRESOLUTION LEVEL-SET APPROACH TO SURFACE FUSION

Augusto Sarti and Stefano Tubaro

Dipartimento di Elettronica e Informazione
Piazza Leonardo Da Vinci 32
20133 Milano - Italy

ABSTRACT

In this paper we propose a novel solution to the problem of combining a number of partial reconstructions through a process of 3D "patchworking". Our method is able to seamlessly "sew" the surface overlaps together, and to reasonably "mend" all the holes that remain after surface assembly, which usually correspond to the non-visible portions of the object surface. The approach is based on the temporal evolution of the zero level set a volumetric function, driven by surface curvature and distance from data and works entirely in a multi-resolution fashion.

1. OVERVIEW AND GLOBAL STRATEGY

A complete 3D model of an object is often obtained by combining a number of partial reconstructions (surface patches) through a process of 3D "patchworking". This operation is based on a preliminary *registration* phase, in which all the available surface patches are correctly positioned and oriented with respect to each other, followed by the actual *fusion*, which consists of merging all surface patches together into a single and closed surface. Notice that the fusion, as described here, has a twofold purpose: to seamlessly "sew" the surface overlaps together, and to reasonably "mend" all the holes that remain after surface assembly, which usually correspond to the non-visible portions of the object surface.

This "atlas" approach to 3D modeling is suitable for modeling solutions based on depth estimation (each depth map is a partial reconstruction), such as image-based depth estimation techniques, range cameras, and laser-scanners. Such methods, in fact, produce depth maps in explicit form, each of which could be made of a several non-connected surface patches, because occlusions and self-occlusions generate depth discontinuities [1]. Such surfaces usually need a lengthy assembly process in order to become a complete and closed surface.

The first step of the assembly process is registration (see Fig. 1), which consists of determining the correct relative positions and orientations of all surface patches. One rather standard strategy is the Iterative Closest Point (ICP) [2] algorithm, which consists of minimizing the mean square distance between overlapping portions of the surface, using an iterative procedure. The focus of this contribution, however, is the fusion process, which we illustrate in the next Section.

2. SURFACE FUSION

A simple way to represent a 2D closed surface γ embedded in the 3D space is in implicit form:

$$\gamma = \{\mathbf{x} \mid \psi(\mathbf{x}) = 0\} \quad ,$$

where $\psi(\mathbf{x})$ is a volumetric function whose absolute value in $\mathbf{x}$ is given by d, which is defined as the distance between $\mathbf{x}$ and the surface, and its sign depends on whether the point $\mathbf{x}$ is inside or outside the surface. Adopting the signed distance as a volumetric function is known to simplify the computation of the surface's differential properties of orders 1 and 2:

- the surface normal can be computed as the gradient $\nabla \psi(\mathbf{x})$ and is a unit vector;
- the surface curvature can be computed as a divergence of the form $\nabla \cdot \nabla \psi$.

We can now redefine the above implicit surface as a temporally evolving levelset, according to the following update equation

$$\psi(\mathbf{x}, t + \Delta t) = \psi(\mathbf{x}, t) - \nabla \psi(\mathbf{x}, t) \cdot F(\mathbf{x}) \cdot \Delta t \quad ,$$

where $F(\mathbf{x})$ is a velocity term that can be defined and controlled in order to steer the levelset toward a desired shape. Our fusion process is, in fact, based on the temporal evolution of the zero level set a volumetric function [3,4] of the above form, as driven by two contrasting needs: that of following the motion by curvature and that of honoring the data (partial reconstructions).

A surface is said to follow the motion by curvature when the velocity field that describes the surface deformation is normal to the surface itself and its magnitude is proportional to the local curvature (with sign). Indeed, if the motion were purely by curvature, a surface would tend to deflate completely and disappear, while becoming progressively smoother and smoother (Fig. 1 above). The need to honor the available range data prevents this complete implosion from taking place (Fig. 1 below).

0-7803-6725-1/01/$10.00

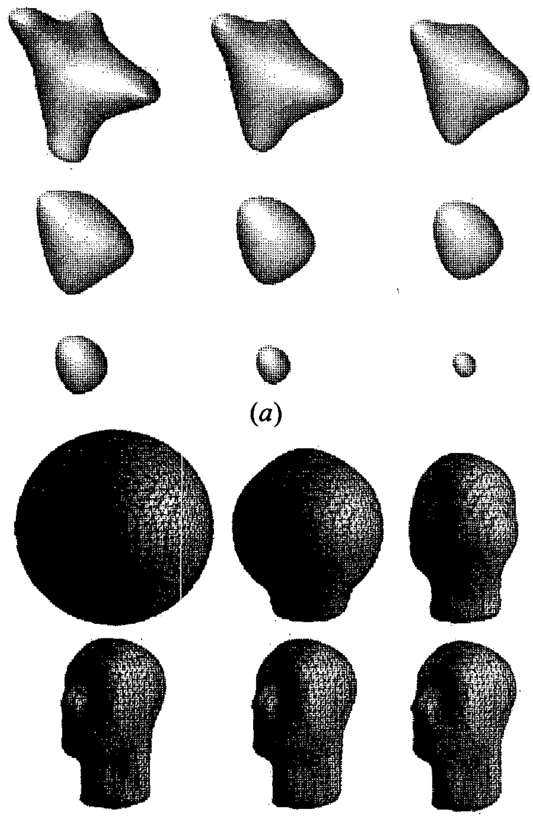

Fig. 1: Motion by curvature: without contrasting action: the level set collapses into a point (*a*). Level set implosion with contrasting action (*b*).

In order to implement this implosion-inhibition mechanism, we need to redefine the velocity field associated to the update equation that describes the zero level-set propagation. This velocity is bound to be orthogonal to the propagating front, and its amplitude is set to

$$F(\mathbf{x}) = F_1(K) + \alpha \frac{K_M}{K} F_2(d)$$

where K is the local curvature of the level-set, K_M is the local curvature of the facing surface; d is the distance (with sign) between the propagating front and the surface patch; and α is a parameter that balances local smoothness and proximity to the data. Indeed, this formulation assumes that only one surface patch is facing the propagating front.

Notice that the above definition of velocity holds valid only for the points that lie on the propagation front, therefore we need to extend its validity in the whole volume (or at least in the sorrounding points of the surface). The extension of this function needs to be done consistently with the front propatation, meaning that the levelset should evolve with no self-collisions. This can be done quite easily [3] as follows: given a generic point $\mathbf{x}$ not lying on the surface γ, we can search for the point $\mathbf{y}$ on γ that lies the closest to $\mathbf{x}$ and let $F(\mathbf{x})=F(\mathbf{y})$.

Given a point on the propagating front, the distance from a surface patch is computed from the orthogonal projection of that point onto the surface patch itself (Fig. 2). If no point on the surface patch faces the point on the level-set orthogonally, then the distance function d is computed from the closest point on the border of the patch (within a pre-assigned range).

When more than one surface patches are facing the propagation front, then the distance function d is computed using the distances from the point on the propagating front and all the orthogonal projections onto the surface patches that face it; the surface orientation; the closeness to the border of the patch (the reliability tends to decrease in the proximity of extremal boundaries); and the mutual occlusion between surface patches.

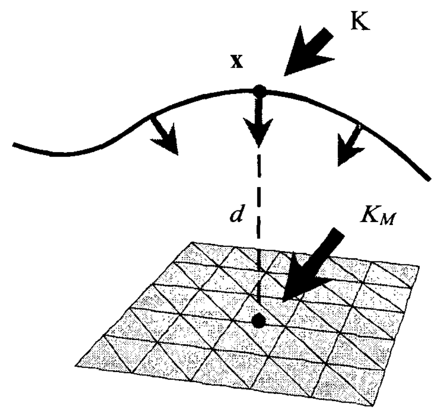

Fig. 2: Given a point on the propagating front, its distance from a surface patch is computed from the orthogonal projection onto the target triangle mesh.

3. KEY FEATURES

Our solution to the surface fusion problem exhibits a number of desirable characteristics. One of its most appealing features is the fact that it is very robust against topological complexity. In fact, a level-set of a volumetric function is adequate for describing multiple objects and arbitrary topologies (see Fig. 3).

The most important aspect of our solution is its modest computational requirements. In order to obtain high performance at low computational cost, besides updating the volumetric function just in a narrow region around the zero level-set (narrow-band implementation), we operate in a multiresolution fashion (see Figs. 4 and 5). This can be achieved by starting with a low-resolution voxset (e.g. a cuberille with 10 voxels per side) and letting the front settle down. Then we break down the voxels around the propagating front and resume the front propagation. The operation continues until the final resolution is reached.

A key aspect of this process is in the fact that the velocity field that drives the implosion of the level set can be pre-computed on the octree data structure that best fits the available range data.

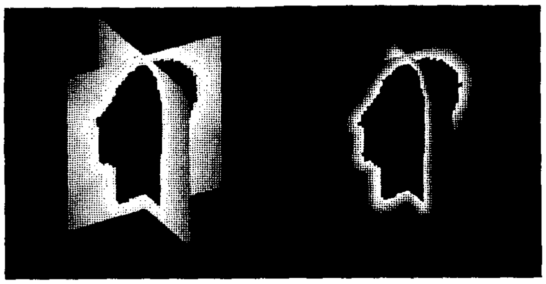

Fig. 3: Complete (left) and narrow-band (right) volumetric representations of an object.

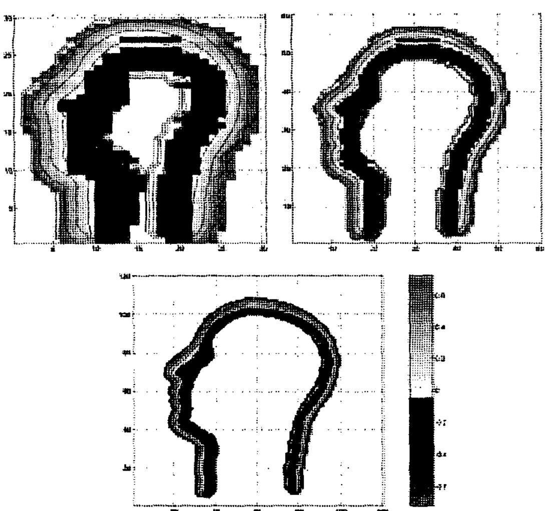

Fig. 4: Multiresolution progression of the voxset where the volumetric function is defined.

The resulting model is bound to be a set of closed surfaces, therefore all the modeling "holes" left after mosaicing the partial reconstructions are closed in a topologically sound fashion. In fact, those surface portions that cannot be reconstructed because they are not visible, can sometimes be "patched up" by the fusion process. This ability to "mend" the holes can also be exploited in order to simplify the 3D acquisition session, as it allows us to skip the retrieval of some depth maps (see Fig. 6 to 8).
An interesting aspect of our fusion method is in the possibility to modify the surface characteristics through a processing of the volumetric function. For example, filtering the volumetric function results in a smoother surface model. Finally, the method exhibits a certain robustness against orientation errors, as the non perfect matching of surface borders can be taken care of by the fusion process through a careful definition of the distance function used in the specification of the volumetric motion field.

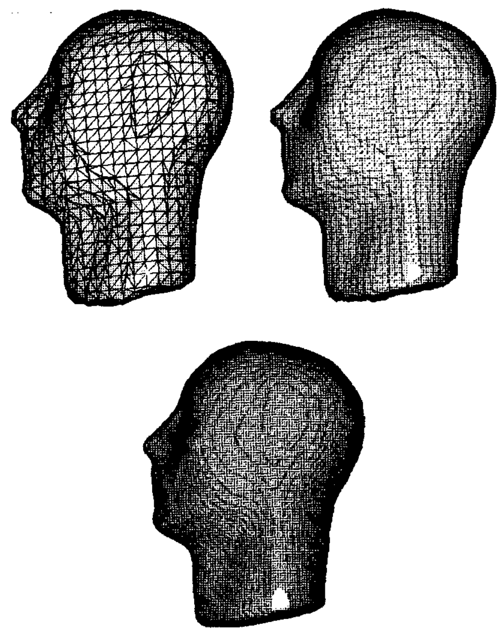

Fig. 5: Multiresolution progression of the zero level-set of the volumetric function.

4. EXAMPLE OF APPLICATION

In order to test the effectiveness of the proposed technique, we applied it to a variety of study cases. A particularly interesting experiment was conducted on an object with a particular topology (a bottle with a handle) that could easily create problems of ambiguities. Any traditional surface fusion approach would, in fact, encounter difficulties in deciding how to complete the surface in the missing regions. Furthermore, besides exhibiting self-occlusion problems, this object puts the multi-resolution approach under a severe test. We acquired six depth maps (Fig. 6) and assembled them together using an ICP algorithm. The result was an incomplete model with some accuracy problems in the overlapping regions (at the boundaries of the depth maps). The resulting front evolution is shown in Fig. 7, while the (topologically correct) final model is displayed in Fig. 8.

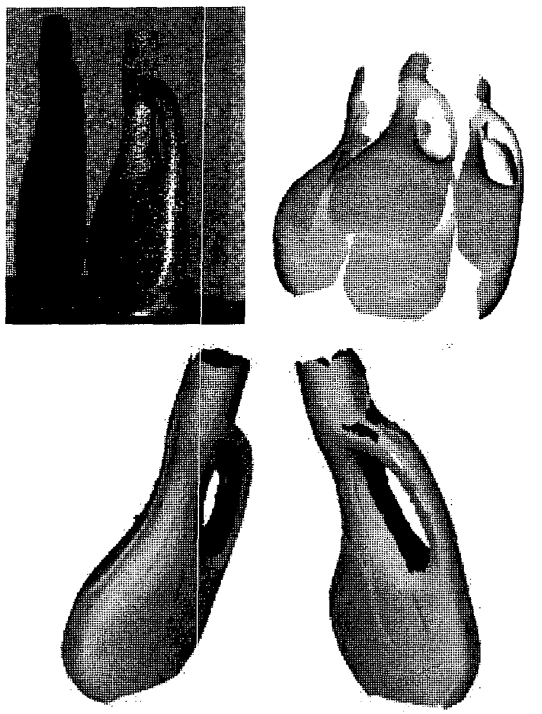

Fig. 6: One of the original views (top left). Six unregistered surface patches obtained with stereometric techniques (top right). Two views of the assembled surface patches after registration (bottom): notice the creases due to a non-perfect model overlapping, and the presence of holes in the global model.

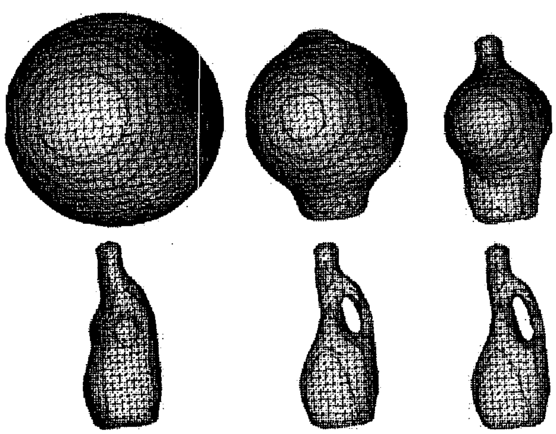

Fig. 7: Level set implosion.

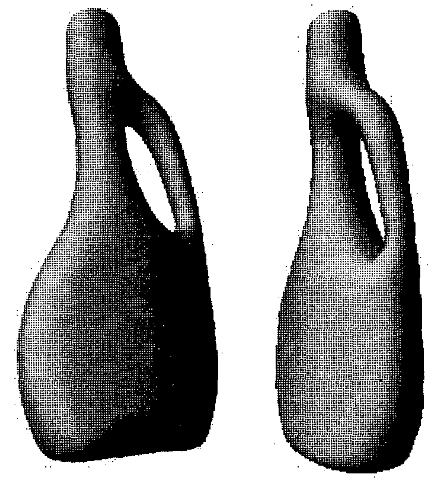

Fig. 8: Final 3D model.

5. CONCLUSIONS

In this paper we proposed a novel solution to the problem of combining a number of partial reconstructions through a process of 3D "patchworking". Our method proved able to seamlessly "saw" the surface overlaps together, and to reasonably "mend" all the holes that remain after surface assembly. Our approach was based on the temporal evolution of the zero level set a volumetric function, driven by surface curvature and distance from data and works entirely in a multi-resolution fashion.

6. REFERENCES

[1] F. Pedersini, A. Sarti, S. Tubaro: "Visible Surface Reconstruction with Accurate Localization of Object Boundaries", *IEEE Tr. on Circuits and Systems for Video Technology*, Vol. 10, No. 2, March 2000, pp. 278-291.

[2] P.J. Besl, N.D. McKay, "A method for registration of 3-D shapes". IEEE Tr. on Pattern Analysis and Machine Intelligence. Vol. 14, No. 2, 1992, pp. 239-256.

[3] R. Malladi, J.A. Sethian, B.C. Vemuri, "Shape Modeling with Front Propagation: A Level Set Approach". IEEE Tr. on Pattern Analysis and Machine Intelligence, Feb. 1995, Vol. 17, No. 2, pp. 158-175.

[4] S. Osher, J.A. Sethian, "Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations". J. Comput. Phys. Vol. 79, pp. 12-49, 1988.