Shape From Silhouette: Towards a Solution for Partial Visibility Problem

B. Michoud[1], E. Guillou[1] and S. Bouakaz[1]

[1] Lyon Research Center for Images and Intelligent Information Systems (LIRIS)& University Lyon 1, France

Abstract

Acquiring human shape is a prerequisite to many applications in augmented and virtual reality, as well as in computer graphics and animation. The acquisition of a real person must be precise enough to have the best possible (realistic) rendering. To do so in real time, "Shape-from-silhouette" (SFS) methods are used. One limitation of these methods is that the acquired subject must be visible from all the camera filming it. If not, some parts of the object are not reconstructed in 3d. This paper presents a modified SFS algorithm, that extends the 3d reconstruction space. Our extension allows to build an estimation of an object's 3d shape even if it comes out of sight from one or more cameras.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism − Virtual reality I.4.8 [Image Processing and Computer Vision]: Scene analysis − Shape

1. Introduction

We wish to perform the real-time insertion into a virtual environment of a person filmed by several calibrated cameras. As the insertion must be as realistic as possible, it is important to model the photometrical and geometrical interactions precisely. To realize this, we need to know a 3d representation of the person.

The literature proposes methods for human shape acquisition. One of the most popular is shape-from-silhouette (SFS). More recently, several SFS-based algorithms allow acquisition and rendering of human shape in real time.

SFS methods compute a shape estimation of an object (called its Visual Hull, noted VH) from its silhouette images. Silhouette images are binary information associated with captured images of the objects where 0 represents the background and 1 stands for the object itself.

When filmed, it is difficult for a moving actor to stay completely visible from all cameras at any given time. Hence, parts of the body will frequently become non visible from one or more cameras, and won't be reconstructed by any of the already published SFS methods. As shown in Figure 1, a significant portion of the actor is not seen in silhouette #3

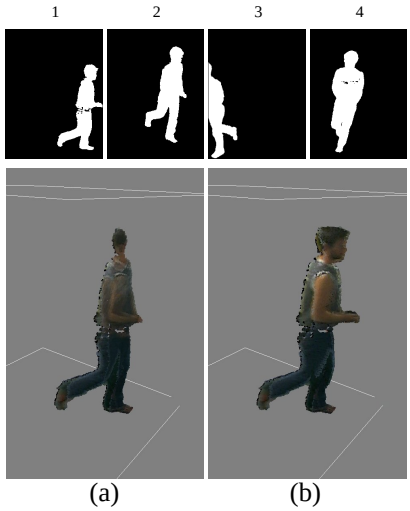

Figure 1: *Differences in 3d shape reconstruction when objects do not project themselves onto all images: (a) using already published SFS methods; (b) using our modified SFS algorithm. Voxel coloring, done using backward raytracing, is only given for image comprehension.*

and is subsequently not reconstructed by previous SFS methods (Figure 1.a).

In this paper, we propose an extension to SFS methods, which removes this limitation, as long as the acquired object is partially visible by all the cameras. After a short summary of SFS principles and background works, we will introduce our reconstruction method which makes it possible to extend the acquisition space of an object O. Then we will discuss on the results obtained. Finally we will present some perspectives for this work.

2. Shape From Silhouette Principles

SFS methods are commonly used to build an object's 3d estimation. The formalism for SFS methods and VH reconstruction algorithms were first introduced by A. Laurentini [Lau94]. The methodology is as follows:

Let object O be acquired by n cameras cam_i, M_i be the projection matrix for camera cam_i, and finally let I_i be the image given of the object by camera cam_i from which a silhouette image S_i may be computed.

If a 3d point P is located in the volume of O then it projects itself onto all silhouette images:

$$\forall i = 1,...,n \ , \ \exists p_i \in S_i \ , \ p_i = M_i.P.$$

where p_i is the projection of P onto the silhouette image S_i.

The VH of O will then be defined as the volume containing all the 3d points that project themselves onto all silhouette images S_i. There are mainly two ways to compute an object's VH, which we now detail.

Surface-Based Approach

An object's VH computed from a set of n silhouette images, is computed as the intersection of all silhouette cones. These cones are defined by the projection, in 3d space, of the silhouette contours through the associated camera's center of projection. This definition gives us a direct computation algorithm. VH is described by a set of 2d patches, each patch is defined as the intersection between the surfaces of the silhouette cones. On the one hand, algorithms based on this approach work in real-time [MBM01, LMS03]. On the other hand, the results are not usable to compute volumetric information required to match generic human models (used for human pose estimation and movement interpretation).

Volume-Based Approach

An equivalent approach defines an object's VH as the maximum volume that projects itself onto all silhouettes of O [Lau94]. Based on this definition, the mostly used algorithm [CKBH00, HLS04] computes an estimation of O's VH with a set of voxels: the 3d region of interest is split into m voxels V_j where $j = 1,...,m$.

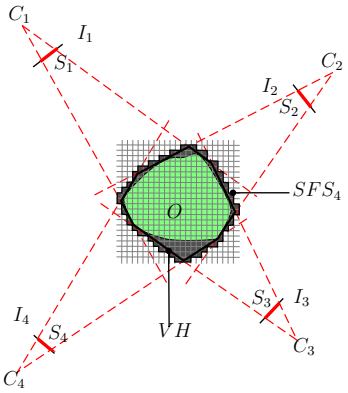

Figure 2: *2d representation of an object O filmed with 4 cameras, the corresponding VH and its voxel-based estimation.*

Let v_{ij} be the set of pixels in I_i, that are projections of voxel V_j:

$$v_{ij} = (M_i.V_j) \cap I_i.$$

We define nb_j as the number of silhouettes in which V_j projects itself:

$$nb_j = Card\{v_{ij} \ , \ v_{ij} \cap S_i \neq \emptyset\}.$$

Let SFS_n be the voxel based Shape From n Silhouettes, which estimate O's VH. If a voxel V_j projects itself onto all silhouettes ($nb_j = n$), then it belongs to O's VH:

$$SFS_n = \bigcup_{j=1}^{m} (V_j \ , \ nb_j = n).$$

where n is the number of cameras used (see Figure 2).

Limitations

In several SFS methods, the region of interest in 3d space, where an object is reconstructed, is given by the intersection of the cameras' vision cones. Limitations are:

- the acquisition area cannot be extended beyond the vision cones intersection, especially when there are many cameras,
- it is difficult to force objects to stay visible to all cameras at any time, since objects can be dynamic and some parts of them may leave the focus of the cameras. Those parts will not be reconstructed.

However, the missing information could often be extracted from the other cameras. We will use this to account for areas not seen by some of the cameras.

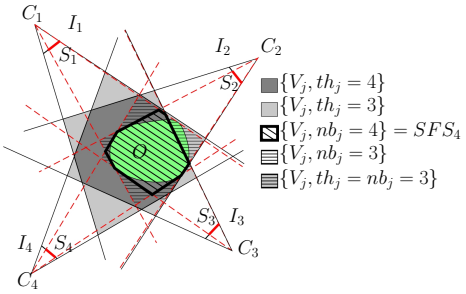

Figure 3: *2d representation of an object O and its estimation using SFS_4. The potential voxels that could extend the shape of O are those for which $th_j = nb_j = 3$.*

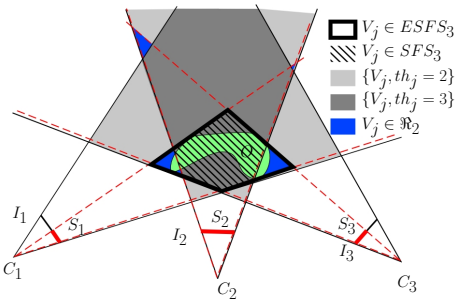

Figure 4: *2d representation of O, its volume estimation from SFS_3 and $ESFS_3$ with $n_{min} = 2$. We can see that $ESFS_3$ is more reliable than SFS_3.*

3. Contributions

To circumvent these SFS limitations, we introduce th_j as the number of images onto which a voxel V_j projects itself :

$$th_j = Card\{v_{ij}\ ,\ v_{ij} \cap I_i \neq \emptyset\}.$$

Then we compute the VH in the usual way and add all parts that:

- projects onto all possible silhouettes: if a voxel V_j belongs to the volume of O then it projects itself onto th_j images and nb_j silhouettes; hence $th_j = nb_j$.
- are connected to the previously computed VH. We use the connex property of the filmed object, to choose which information is liable to extend the object's VH.

We now detail the method. Our goal is to find which voxels V_j belong to O's volume. For each voxel V_j we compare th_j to nb_j:

- If $th_j \neq nb_j$ then $V_j \notin O$'s volume;
- else V_j potentially belongs to O's volume (Figure 3).

The set of all such voxels, may be split into n subsets R_i :

$$R_i = \{V_j\ ,\ nb_j = th_j = i\}.$$

We notice that

$$SFS_n = \bigcup_{V_j \in R_n} V_j.$$

Hence, to extend SFS_n we choose voxels from subsets R_k with $k \in [n_{min}, \cdots, n-1]$. Let $\Re_{n_{min}}$ be the union of all R_k:

$$\Re_{n_{min}} = \bigcup_{k=n_{min}}^{n-1} R_k.$$

Now we use the filmed object's connex property: the 3d object to reconstruct is connex, so it's 3d reconstruction is also connex.

$\Re_{n_{min}}$ is the union of L connex components noted c_l where $l = 1, ..., L$. To satisfy the connexity of the reconstruct volume, we choose the connex components from $\Re_{n_{min}}$ which are connected to SFS_n (as shown in Figure 4).

Let $C_{n_{min}}$ be the set of connex components of $\Re_{n_{min}}$ connected to R_n:

$$C_{n_{min}} = \bigcup_{l=1}^{L} (c_l\ ,\ connex(c_l \cup R_n)).$$

Then we introduce $ESFS_n$ an extension of SFS_n:

$$ESFS_n = SFS_n \cup C_{n_{min}}.$$

The $ESFS_n$ behaves exactly as the SFS_n when the person is well inside all visual cones, and it is likely to cause additional errors only when the person is out to the field of view limits. The additional error depends on n_{min} value:

- If $n_{min} = n-1$, then $ESFS_n$ could complete the SFS_n reconstruction, with information seen on $n-1$ cameras, and that is the best precision for strictly less than n cameras.
- If n_{min} is near 1, then the acquisition space proposed by $ESFS_n$ is larger than that for SFS_n. But the reconstructed shape will have a bad precision for the sections of O seen only from n_{min} cameras.

Thus the n_{min} value choice depends on the application of the reconstructed shape.

4. Results

Our extension has been tested on various sets of real data acquired from $n = 4$ calibrated cameras with an image resolution of 320x240 pixels. The best reconstruction precision when some sections of O are out of sight for only one camera is given by $n_{min} = 3$.

Figure 5 shows reconstruction results for a complex object. Having a partial visibility in silhouettes #1, #3 and #4, usual SFS algorithms give only partial reconstructions (see Fig. 5.a). Our reconstruction allow to build a 3d representation for the chair as the parts that are not visible from all cameras at the same time vary from one camera to the next. The chair's legs are not represented in both methods as it is invisible from most of the cameras.

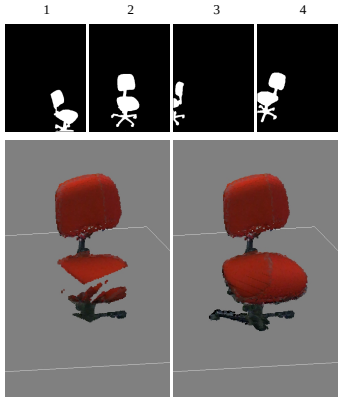

Figure 5: *Voxel based shape estimation for a complex object: (a) shape computed by original SFS algorithm; (b) using our extended SFS algorithm.*

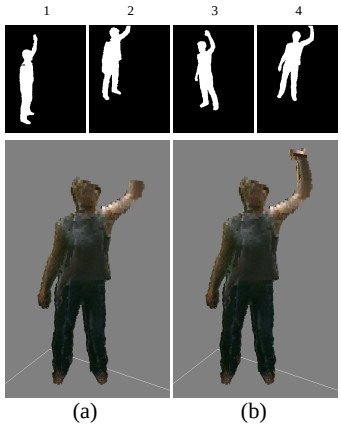

Figure 6: *Voxel based shape estimation for a complex object: (a) shape computed by original SFS algorithm; (b) using our extended SFS algorithm.*

Figures 1 and 6 show the results obtained when acquiring a moving human actor. In both cases, our extended method adds new valid information to the estimated shape.

Our algorithm performs very closely to the original SFS. Indeed, only two steps were added:

1. The computation of the object's th_j is carried out only once as a preprocess step of the algorithm. As a matter of fact, they depend only on the cameras' parameters which are constant during the acquisition process;
2. The computation of 3d connexity: for each frame, we traverse the sets R_n and R_k. The traversal time is insignificant compared to the computation time of voxels' projection onto each silhouette.

The experimental implementation of our extension computes approximately 60 object's estimations per second (for a set of 128^3 voxels and $n_{min} = 3$), while our implementation of actual SFS provides 65 object's estimations per second. This is suitable for applications claiming real time.

5. Conclusion and Future Work

In this paper, we have proposed an extension of Shape from Silhouette algorithm which makes it possible to extend the acquisition's space of an object's shape compared to the one obtained with common SFS algorithms. Apart from the camera's calibration problem, our only assumption is that the object must be mainly visible by all cameras.

The shape's estimation from our method contains the one that could be obtained with actual SFS. Our extension can estimate the form of all parts of an object that are not visible in one or several cameras, as long as those sections are completely visible by all other cameras. Even with the extension, the reconstruction algorithm works in real time.

We are currently working on the characterization of reconstruction error depending on the number of used cameras. This method is used for motion tracking in real time. Aiming to work with high frequency cameras, we are working on a GPU implementation of our algorithm. Our experimental implementation already uses Projective Texture Mapping methods to computes voxel's projections onto silhouettes. It is now necessary to develop an efficient algorithm for 3d connex components traversal on GPU.

References

[CKBH00] CHEUNG K. M., KANADE T., BOUGUET J.-Y., HOLLER M.: A real time system for robust 3d voxel reconstruction of human motions. In *Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)* (Juin 2000), vol. 2, pp. 714 – 720.

[HLS04] HASENFRATZ J.-M., LAPIERRE M., SILLION F.: A real-time system for full body interaction with virtual worlds. *Eurographics Symposium on Virtual Environments* (2004), 147–156.

[Lau94] LAURENTINI A.: The visual hull concept for silhouette-based image understanding. *IEEE Trans. Pattern Anal. Mach. Intell. 16*, 2 (1994), 150–162.

[LMS03] LI M., MAGNOR M., SEIDEL H.: Hardware accelerated visual hull reconstruction and rendering. In *Graphics Interface* (2003), pp. 65–71.

[MBM01] MATUSIK W., BUEHLER C., MCMILLAN L.: Polyhedral visual hulls for Real-Time rendering. In *12th Eurographics Workshop on Rendering Techniques* (2001), pp. 115–126.