LDI 평면 이동에 의한 이미지 기반 Surfel 복원
(Image-based Surfel Reconstruction by LDI Plane Sweeping)

이 정 [†] 김 창 헌 [††]
(Jung Lee) (Chang-Hun Kim)

요 약 본 논문은 입력받은 다수 이미지의 Visual Hull을 이용하여 Surfel 기반의 물체를 복원하는 새로운 방법을 제안한다. Surfel은 포인트 집합면을 효과적으로 근사하여 나타내는 포인트 표현 단위이다. LDC(Layered Depth Cube) Surfel 샘플링 방식에 입력 이미지로부터 근사되는 형상을 대표하는 Visual Hull의 개념을 연동하여 이미지로부터 물체의 Surfel 기하 정보를 생성할 수 있다. Surfel 표현 방식은 기존의 폴리곤 기반 방식에 비해 상대적으로 적은 메모리를 소모하고 LDC 해상도를 자유롭게 바꿀 수 있기 때문에, 대상 물체를 복원하는 품질을 조절 가능하며, 주어진 메모리 자원에 대해서 최대의 품질을 가진 결과를 얻을 수 있다.

키워드 : surfel, visual hull, LDC 샘플링

Abstract This paper proposes a novel method that reconstructs a surfel-based object by using visual hull from multiple images. The surfel is a point primitive that effectively approximates point-set surface. We create the surfel representation of an object from images by combining the LDC(Layered Depth Cube) surfel sampling with the concept of visual hull that represents the approximated shape from input images. Because the surfel representation requires relatively smaller memory resources than the polygonal one and its LDC resolution is freely changed, we can control the reconstruction quality of the target object and acquire the maximal quality on the given memory resource.

Key words : surfel, visual hull, LDC sampling

1. Introduction

Recent advances in scanning devices make it easy to acquire hundreds of millions of points. So the geometric objects have so many polygons and it is difficult to manipulate them. When rendering these large and dense polygonal meshes, several primitives may often overlapped in one pixel. This redudancy cannot be ignored in time-critical applications. Additionally, triangulation may corrupt the geometry by forcing connections between scanned points.

So the point-based geometry has been focused, because it does not require the point connectivity information and less memory resources. Grossman and Dally[1] sampled point data sets from geometric models for their point rendering. They also addressed the issues of sampling rate and gaps in the point-rendered images. The surfel rendering scheme[2] has been proposed to render large sets of points. A surfel is a point primitive that has a position, a normal vector, and a disk radius. By using an octree-based structure and forward warping technique, surfel objects are rendered accurately at interactive frame rates.

In order to create continuous surfaces from irregularly spaced point samples, splat-based approaches

---

· This research is supported by Ministry of Culture, Sports and *Tourism (MCST) and Korea Creative Content Agency (KOCCA) in* the Culture Technology (CT) Research & Development Program 2009.

· This work is supported by the Second Brain Korea 21 Project.

† 정 회 원 : 고려대학교 BK21 소프트웨어 사업단 교수
airjung@gmail.com
†† 종신회원 : 고려대학교 컴퓨터통신공학부 교수
chkim@korea.ac.kr

논문접수 : 2009년 1월 14일
심사완료 : 2009년 8월 27일

정보과학회논문지: 소프트웨어 및 응용 제36권 제11호(2009.11)

have been suggested. QSplat[3] replaces point primitives by ellipses or rectangles when projected onto image space. A hierarchical bounding sphere structure facilitates time-critical rendering. Surface splatting[4] applies EWA(Elliptical Weighted Average) texture filtering in object space and has achieved the image of high quality including antialiasing capabilities.

Alexa et al.[5] have proposed a new method to compute and render point-set surfaces using a MLS (Moving Least Squares). A new explicit definition of point-set surfaces[6] has also been introduced to represent various point primitives such as splats and surfels.

As an application of point geometry, Adams and Dutre[7] proposed CSG(Constructive Solid Geometry) operations on free-form solids bounded by surfels. By resampling surfels that intersect with the surface of another solid, they maintain sharp boundaries. They have shown that surfels provide a useful tool for interactive editing of free-form solids.

Volumetric carving algorithms[8,9] recover the voxel representation of a shape from multiple images. A complete 3D model is created by iteratively eliminating, voxels from an initial large volume covering the target object. They called this elimination process ‘carving’. The decision to carve a voxel is determined by its color-consistency on its visible images. They introduce plane sweeping technique that frees the carving process from vision constraints such as occlusion or parallax. But they have some serious problems in voxel quantization and large memory resource. When the resolution of an initial volume is lower than that of source images, the final rendered images are so blurred and lose the object details. And the voxels of the initial bounding volume has been arranged regularly, including so many potential empty ones, the memory resource is wasted uselessly in most cases.

This paper proposes a novel method of creating surfels from multiple images. To reconstruct them accurately, we combine the LDC surfel sampling scheme with the concept of the visual hull[10]. And we introduces LDI(Layered Depth Image) plane sweeping technique instead of the static bounding volume, we reduce the amount of memory required, thus allowing us to increase the sampling resolution.

The remainder of this paper is organized as follows. Section 2 introduces some definitions and assumptions. Our modified surfel sampling method is explained in Section 3. Section 4 shows some experimental results and discussion. Then we conclude this paper in Section 5.

2. Definitions and Assumptions

2.1 Image Constraints

As with image-based modeling approaches, we start with calibrated images that captured from arbitrarily distributed cameras. That is to say, we already know the projection matrices of all input images. To evaluate the efficiency and accuracy of our method, we use synthetic images with a segmented background (see Figure 1).

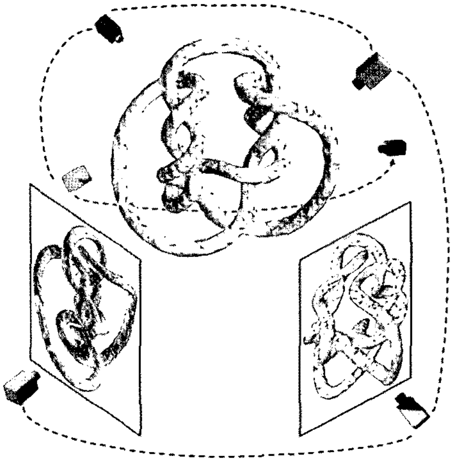


Figure 1 Input Image Acquisition

2.2 Image-based Surfel Representation

The surfel is an effective point primitive for rendering the high-quality images of point geometry without requiring explicit connectivity between adjacent points. Each surfel is often rendered as a circular disk which has attributes including its position (p_x, p_y, p_z), normal vector $n(n_x, n_y, n_z)$, and disk radius r. Surfels and their attributes have been usually sampled from the existing 3D models. In this paper, however, we will generate surfels from multiple images and Figure 2 shows our scheme of point-set object reconstruction from images.

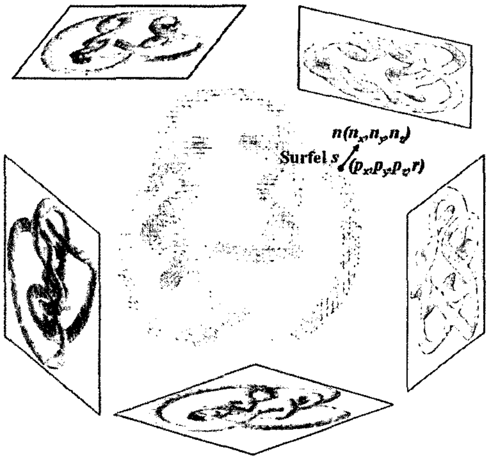


Figure 2 The Concept of Surfel Sampling from Multiple Images

3. Visual Hull Based Surfel Sampling

We will now explain in detail how to sample surfels and their attributes from multiple images. In the original surfel sampling method[2], surfels are sampled by using a ray-surface intersection test on polygonal model. So it is a major problem that we don't know the surfaces of a target shape when we sample surfels from images. In the image-based approaches, the approximation of the true surfaces is called the visual hull. So we create surfels that are assumed to be on the surfaces of the visual hull using color-consistency constraint (see Figure 3).

Figure 4 shows the processing steps. First, we create an LDC structure which consists of 3 LDIs with user-specified sampling resolution m. [2,11] Then we sweep each LDI plane in 6 directions parallel to the x, y, and z axes, not ray casting performed in the previous approach [2]. As each LDI plane moves, some cameras are activated to be included in color-consistency test. For each LDI pixel position, the color-consistency test finds whether a surfel is sampled at that position. After all the space has been swept, the disk radii and normal vectors of the sampled surfels are estimated and the reconstructed surfel objects are finally acquired.

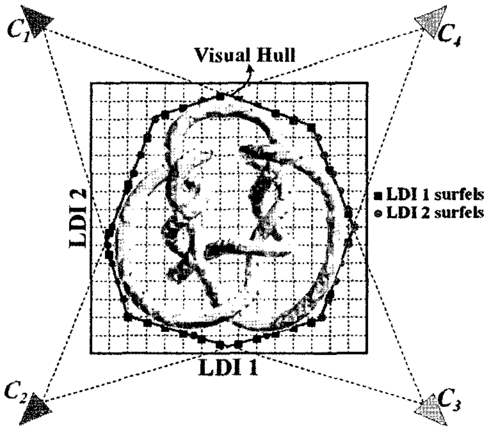


Figure 3 A 2D Illustration of Surfel Sampling Using a Visual Hull

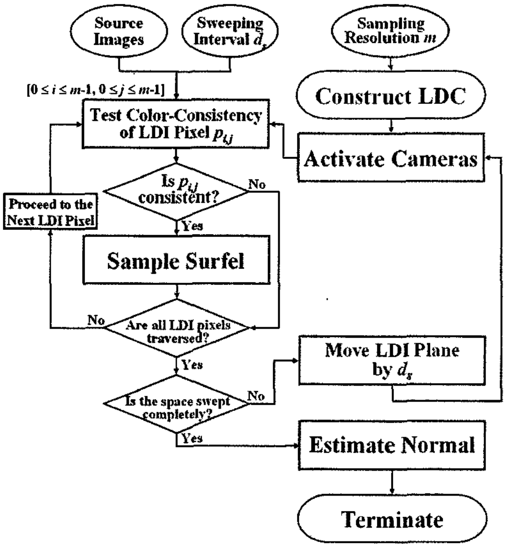
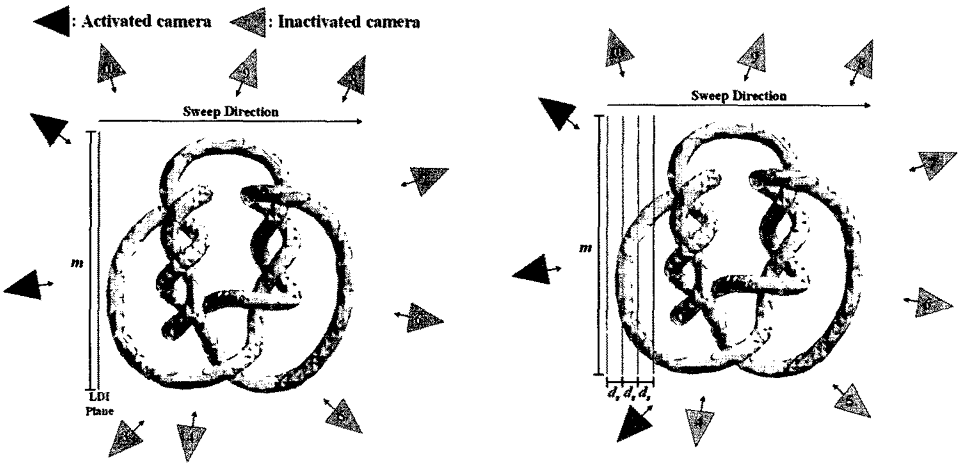


Figure 4 Algorithm Overview

3.1 LDI Plane Sweeping

Before testing the color-consistency of each LDI pixel for carving, we must determine the set of images in which the current LDI pixel position is visible - not occluded by the other LDI pixels. This is called visibility computation. We introduce a novel method that combines LDC sampling scheme with the plane sweeping method[9] and is called LDI plane sweeping. The detailed process is as follows.

In this process, we move the sweep planes, called LDI planes in this paper, in positive and negative directions parallel to the *x*, *y*, and *z* axes. In the previous method[9], the sweep planes were sligthly moved along each direction by the static voxel grid spacing. But we can move the LDI planes by a user-specified sampling interval d_s (see Figure 5).



(a) (b)

Figure 5 LDI Plane Sweeping Overview

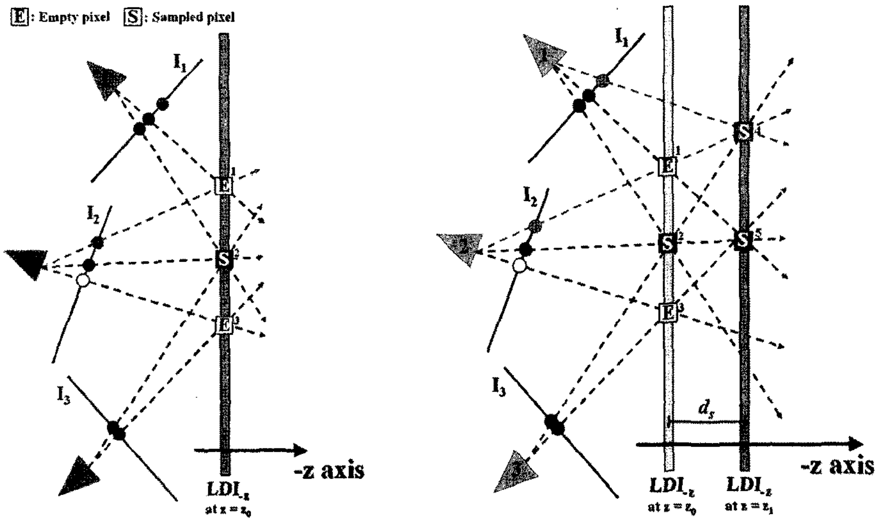
When each LDI plane is moved, we investigate the cameras to be activated in color-consistency test. As LDI plane moves, each camera is activated to be included in the color-consistency test, if it is behind the LDI plane and the angle between its viewing direction and sweep direction is within a threshold θ. To avoid the problem of blurring in the image capturing pipeline, we set the threshold θ to 60°. As seen in Figure 5(a), only Camera 1 and 2 are activated. Camera 3 is additionaly activated as the LDI plane passes it (see Figure 5(b)). Although Camera 10 is also passed by LDI plane, it cannot be activated because its viewing direction is too far from the sweep direction. By adjusting d_s appropriately, we can control the density of the sampled surfels and avoid the lack of precision introduced by the previous voxel carving methods [8,9].

3.1.1 Color-consistency Constraint

After the camera activation, we sample surfels at the LDI pixels that are assumed to be on the surfaces of visual hull. As widely known, visual hull represents the maximal shape and appearance computed from images. Seitz et al. introduces color-consistency constraint to compute visual hull accurately. They assumed the Lambertian illumination model that shows the pure diffuse reflection only and noticed that the point on the true surface emits the consistent color in any images that capture it. This means that if a certain LDI pixel shows the consistent color in all of its projected pixels on its visible images, it must be a valid surface point. In this paper, all of LDI pixels are tested whether they show consistent colors in the corresponding images of the activated cameras and surfels are sampled at the color-consistent LDI pixel positions. But because they don't show the exactly same color due to the image digitizing process and specular effects existing in real world, we set a threshold value and determine a LDI pixel to be valid if its variation of projected pixel color is within that value.

Figure 6 shows a 2D example of the color-consistency test. In Figure 6(a), LDI pixel 1 is determined not to be on the surface because it has different colors in the images taken by Camera 1 and 2. LDI pixel 2 shows consistent colors in all activated cameras, so a surfel is created at that position. LDI pixel 3 needs no color-consistency test because its projection on the Camera 2 is in the background area and this is definitely outside the target object. From the known property of visual hull, if a certain area is projected onto the background of any image, that is fully outside the target object.

After moving the LDI plane by d_s, the same process is performed on all LDI pixels. Notice that Camera 3 is not included in testing the color-consistency of LDI pixel 4, because this LDI pixel projects on the area already occupied by LDI pixel 2 in the



(a) (b)

Figure 6 2D Example of Color-consistency Test

previous step. This means that LDI pixel 4 is not visible from Camera 3. This property is called ordinal visibility. By testing LDI pixel 4 with Cameras 1 and 2, another surfel is created at that position. For the same reason, camera 2 is not considered in testing LDI pixel 5.

Because this scheme maintains an ordinal visibility that robustly captures all occlusion relations, we can perform both visibility computation and surfel sampling at the same time.

3.1.2 Surfel Attributes Estimation

In this paper, a surfel is rendered as a circular disk with its radius. So we need to compute the radius and normal vector of each reconstructed surfel. To guarantee that adjacent surfels are slightly overlapped and fill the reconstructed surface, we set the surfel radius r to the maximum distance between adjacent surfels in the reconstructed surfel object. The maximum distance is simply computed using LDI pixel spacing dp and the sampling interval ds (see Equation 1).

$$r = \sqrt{2 \times d_p^{\,2} + d_s^{\,2}} \qquad (1)$$

The normal vectors are important to render surfels with an appropriately shaded color. Because we know only their sampled positions yet, the existing normal calculation methods cannot be applied.

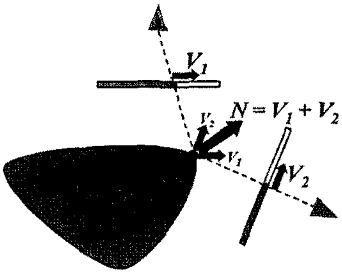


Figure 7 2D Illustration of Surfel Normal Approximation from Images

We notice that normal vector is headed to the outside direction from the object. Using this observation, we investigate the 2D normal vectors of each surfel projected onto all visible images. These 2D vectors are converted into 3D ones. By averaging these 3D vectors computed on each visible image, we can robustly approximate normal vector of each surfel. Figure 7 shows an example of our image-based normal calculation process. We calculate V_1 from its projection to the Camera 1 and V_2 from the Camera 2. Final normal vector N on that surfel is approximated by averaging V_1 and V_2. As seen in this figure, they are guaranteed to be towards outside.

4. Results & Discussion

Our experiments were performed on a system with Pentium IV 2.8GHz CPU and 1GB RAM. We reconstruct each surfel object from 36 images with 512^2 resolution and 0.5 pixel sampling interval. (see Figure 8) The surfels sampled by our method are rendered by PointShop3D 2.0, which is free surfel rendering software written by Zwicker et al.[12].

Figure 9 shows the rendered results of the reconstructed 'Knot' model. They are rendered at the arbitrary viewpoints that are not included in the source images. As seen in this figure, our method works well on an object with complex.

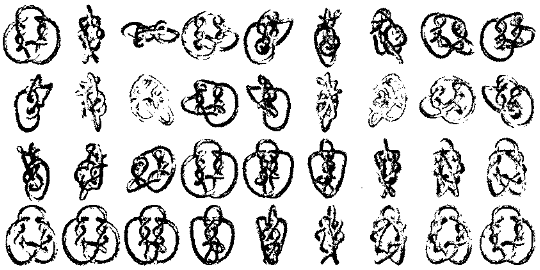


Figure 8 36 Source Images of 'Knot' Model

Figure 10 shows the robustness of our algorithm for the calibration error. If we apply our algorithm to real photos in practice, the calibration error must be the most critical obstacle in the process. For simulating the calibration error condition, we add randomized pixel errors in to the projected LDI pixel positions onto source images. Figure 10(a) is the reconstruction result of a 'knot' model with no projection error. And Figure 10(b~e) are the magnified view of the rectangular area of Figure 10(a) for different reconstruction results with varying projection error. As shown in this figure, the outlier and the incorrectly color-mapped (rendered as white) surfels are increased according to the growing calibration error, but they still represent the target shape well.

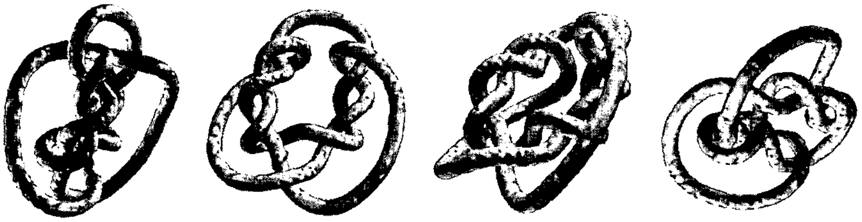
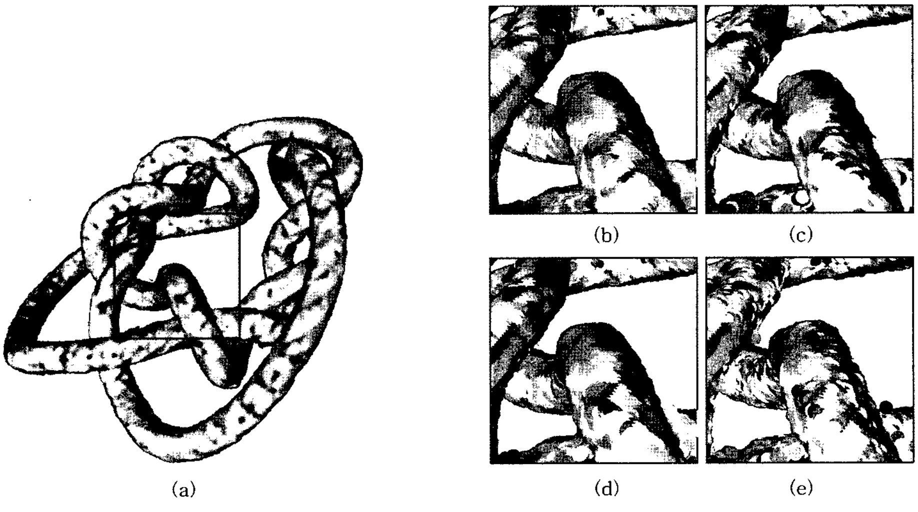


Figure 9 Reconstructed 'Knot'



(a) (b) (c) (d) (e)

Figure 10 Reconstructed 'Knot'. With Randomized Projection Error Bound. (b) No Error, (c) 1 Pixel, (d) 2 Pixels, (e) 4 Pixels

Figure 11 shows reconstruction results for the different sampling intervals. We expected that the visual quality will be enhanced if the sampling interval decreases. But Figure 11(b) shows better appearance and less outlier surfels than Figure 11(c), although Figure 11(c) is more densely sampled. These unexpected results might come from the fact that the outlier surfels also increase because surfels are excessively sampled by the smaller sampling interval. The increase of outlier surfels causes the degradation of visual quality. In Figure 11(a), the falling-off in its quality is natural, because the sampling interval is too large to capture the subtle details of the target object.

Figure 12 shows the limitation of our approach. As seen in this figure, there exist still incompletely recovered surfels, although those regions are visible in some source images. This artifact comes from the inherent property of color-consistency test. Actually, even if a LDI pixel shows the consistent colors in its visible images, it can be outside the target object actually. So outlier surfels showing consistent colors in its visible images are also recovered.

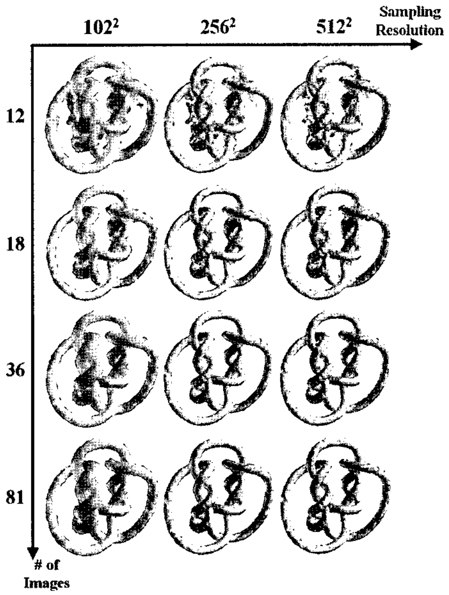


Figure 13 Results for different numbers of source images and sampling resolutions of the LDI

Figure 13 compares the varying results for a 'knot' model by changing the number of source images and the sampling resolution of the LDI. These results are rendered from a new viewpoint which

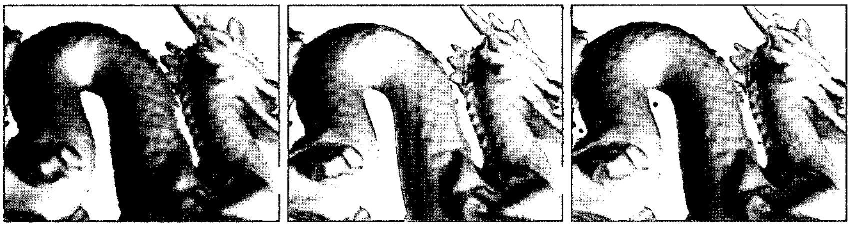

(a) (b) (c)

Figure 11 Reconstructed 'Dragon' for Different Sampling Intervals. (a) 2.0, (b) 1.0, (c) 0.5

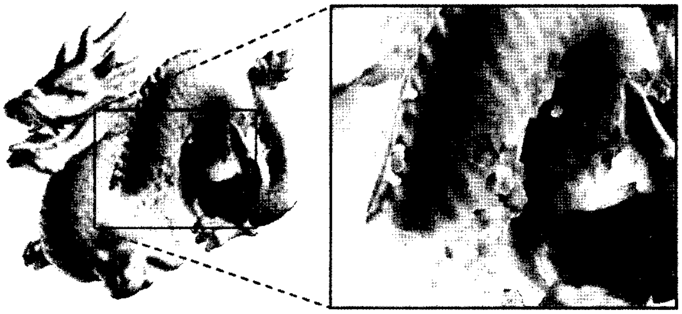


Figure 12 Reconstructed 'Dragon' Model

was not used for generating any of the source images. The quality of the reconstructed shape improves as the number of images and the sampling resolution increase. Note that the visual improvement by changing the number of images appears a little when it reaches a certain value, especially from 36 to 81. An accurate shape and appearance is achieved with 36 images, and little improvement is gained by using more.

By integrating surfel geometry into the previous voxel carving mechanism, we can recover the 3D models with higher quality and render them more quickly. In addition, the Gaussian blending technique of surfel rendering scheme also helps us to reduce the quantization rendering artifacts occurred in the previous regular grid based approaches.

5. Conclusion and Future Work

We have proposed a novel method to sample surfels from multiple images using the visual hull. Our LDI plane sweeping method robustly solves the visibility computation problem in surfel sampling step. By using color-consistency constraint, we obtain surfels that are close to the true surfaces of a target object than those created by previous visual hull approaches. Experimental results show that our method works robustly on various complex objects.

In a future work, we intend to develop a new color consistency criterior to eliminate the outlier surfels in the artifact regions as shown in Figure 10. And it may be possible to reduce some outliers by optimizing our naive surfel sampling procedure.

References

[1] Adams B., Dutre P., "Interactive Boolean Operations on Surfel-bounded Solids," *ACM Transactions on Graphics (SIGGRAPH '03)*, vol.22, no.3, pp.651-656, 2003.

[2] Alexa M., Behr J., Fleishman S., Cohen-Or D., Levin D., Silva C. T., "Computing and Rendering Point Set Surfaces," *IEEE Transactions on Visualization and Computer Graphics*, vol.9, no.1, pp.3-15, 2003.

[3] Amenta N., Kil Y. J., "Defining Point-set Surfaces," *ACM Transactions on Graphics (SIGGRAPH '04)*, vol.23, no.3, pp.264-270, 2004.

[4] Grossman J. P., Dally W. J., "Point Sample Rendering," *In Proc. 9th Eurographics Workshop on Rendering*, pp.181-192, 1998.

[5] Matusik W., Buehler C., Raskar R., Gortler S. J., McMillan L., "Image-based Visual Hulls," *ACM Transactions on Graphics (SIGGRAPH '00)*, vol.19, no.3, pp.369-374, 2000.

[6] Pfister H., Zwicker M., Baar J. V., Gross M., "Surfels: Surface elements as rendering primitives," *ACM Transactions on Graphics (SIGGRAPH '00)*, vol.19, no.3, pp.335-342, 2000.

[7] Rusinkiewicz S., Levoy M., "QSplat: A multiresolution point rendering system for large meshes," *ACM Transactions on Graphics (SIGGRAPH '00)*, vol.19, no.3, pp.343-352, 2000.

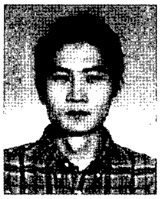
[8] Seitz S. M., Dyer C. R., "Photorealistic Scene Reconstruction by Voxel Coloring," *International Journal of Computer Vision*, pp.151-173, 1999.

[9] Shade J., Gortler S. J., He L., Szeliski R., "Layered Depth Images," *ACM Transactions on Graphics (SIGGRAPH '98)*, vol.17, no.3, pp.231-242, 1998.

[10] Seitz S. M., Kutulakos K. N., "A Theory of Shape by Space Carving," *International Journal of Computer Vision*, pp.199-218, 2000.

[11] Zwicker M., Pfister H., Baar J. V., Gross M., "Surface Splatting," *ACM Transactions on Graphics (SIGGRAPH '01)*, vol.20, no.3, pp.371-378, 2001.

[12] Zwicker M., Pauly M., Knoll O., Gross M., "Pointshop3D: An interactive system for point-based surface editing," *ACM Transactions on Graphics (SIGGRAPH '01)*, vol.21, no.3, pp.322-329, 2002.



이 정

2000년 고려대학교 컴퓨터학과 졸업(학사). 2002년 고려대학교 컴퓨터학과(이학석사). 2006년 고려대학교 컴퓨터학과(이학박사). 2006년 고려대학교 BK21 산업단 연구교수. 2006년~2008년 삼성전자 통신연구소 책임연구원. 2008년~현재 고려대학교 BK21 산업단 연구교수. 관심분야는 컴퓨터 그래픽스, 컴퓨터 비전

김 창 헌

1979년 고려대학교 경제학과 졸업(학사) 1987년 한양대학교 전산학과(이학석사) 1993년 Univ. of Tsukuba 전자정보학과(이학박사). 1979년~1987년 한국과학기술연구소(KIST) 연구원. 1987년~1995년 한국과학기술원 시스템공학연구소 책임연구원. 1989년~1990년 일본 동경공업대학 객원교수 2003년~2004년 UCLA 객원교수. 2005년~2007년 고려대학교 정보통신대학 학장. 1995년~현재 고려대학교 정보통신대학 정보통신공학부 교수. 2008년~현재 한국컴퓨터그래픽스학회 회장. 관심분야는 컴퓨터 그래픽스, 컴퓨터 비전, 유체 시뮬레이션