3D Avatar Modeling and Transmission using Cylinder Mapping

SHIN-ICHIRO TAKAHASHI † ISHTIAQ RASOOL KHAN ‡
MASAHIRO OKUDA †† and MASAAKI IKEHARA †

† Department of Electronics and Electrical Engineering, Keio University
Yokohama, Kanagawa 223-8522, JAPAN
{takahashi,ikehara}@tkhm.elec.keio.ac.jp http://www.tkhm.elec.keio.ac.jp/

‡ Graduate School of Life Science and Systems Engineering Kyushu Institute of Technology,
Kitakyushu, Fukuoka 808-0196, JAPAN, khan@brain.kyutech.ac.jp

†† Department of Information and Media Engineering The University of Kitakyushu,
Kitakyushu, Fukuoka 808-0135, JAPAN
okuda-m@env.kitakyu-u.ac.jp http://vig.is.env.kitakyu-u.ac.jp/

Abstract : In this paper, we propose a real time 3D imaging system. Existing techniques for construction of 3D models require multiple cameras and complex calculations. In our proposed system, the 3D shape is captured by integrating multiple range images obtained simultaneously from different directions. It can accomplish the real time processing by generating the range images quickly and continuously. Moreover, the obtained 3D shape is treated as *"Point Cloud"*, and therefore, the connectivity information of vertices and the calculation of connection topology become unnecessary. Giving importance to transmission efficiency is the advantage of our system. The obtained 3D shape is transmitted to the remote location, and it is reconstructed and depicted immediately. By repeating this procedure continuously, a 3D sequence can be generated in real time. The 3D shape is constructed by integration of multiple range images, causing overlap and annoying artifacts between adjacent images. We apply a cylindrical method to each frame of the 3D sequence to get rid of the artifact and improve the image quality. Our algorithm can remove the invalid points quickly and efficiently, and can be implemented in real time.

Key–Words : Real Time Systems, Stereo Vision, Point Cloud

1 Introduction

In recent years, there has been much interest in generating 3D sequences of real objects. With growing use of 3D images and videos in future, the development of an effective and efficient 3D imaging system is highly desirable. Although many 3D imaging systems have been proposed, most of the conventional methods work off-line at modeling stage. In [1], high-definition 3D images were reconstructed with the system needed 49 cameras. In [2]–[5], these systems accomplished generating the 3D sequence in real time. Most of the systems employed the visual hull method to construct the 3D shape. The methods approximate a rough shape of the target object by using silhouette images and the voxels. Complex calculations and many cameras are needed to get finer shapes and map the texture. In addition, the surfaces of 3D models were represented by polygonal meshes. Generally, the mesh models use vertices and their connectivity information to represent the surface. Thus it requires recalculation of connection topology, and causes increase of the amount of data. Our system employs range images to construct the 3D model without visual hull method. As the surface of 3D model consists of points obtained by integrating multiple range images, the connectivity information is unnecessary. The range image is generated at high-speed by stereo cameras using stereo matching method [6]. Some cameras based on this method have been commercialized. The Digiclops camera by Points Grey Research [7] is one of those, and it can obtain the range images at 10-30 frames per second (fps).

Point based geometry is a well known technique to render the Point Cloud. In our system, we also use a point as a rendering primitive. By using this, vertices on the surface of 3D model are depicted directly without connecting with other vertices, resulting in greater efficiency needed for real time rendering [8].

In certain situations at the modeling stage, it is hard to capture the 3D shape perfectly. Failure to obtain precise camera-calibration or Region Of Interest (ROI) extraction results in overlaps and annoying artifacts especially at the boundaries of the different range images. To address these problems, we apply a cylindrical method to each frame of 3D sequence before rendering stage. This method is similar to the approach introduced in [9]. Because the Point Cloud does not have connectivity information, it is not easy to decide relations of a point to the other. The cylindrical method maps each point of the Point Cloud on the inner wall of a virtual cylinder, which is then warped to a 2D plane. Mapping of 3D points on a plane facilitates handling of the 3D Point Cloud efficiently, and makes the process work in real time.

The paper is organized as follows: Details of our proposed 3D transmission system are explained in Section 2. The procedure of the cylindrical method is described in Section 3. Equipments of our system and some results of produced 3D models which applied the cylindrical method are presented in Section 4. Conclusions and future works are briefly expressed in Section 5.

2 Real Time 3D Transmission System

2.1 Overview

Our proposed system consists of a Client and Server setup shown in Fig.1. The server consists of some PCs with stereo camera and a PC for controlling. Roles of the server are generation of range images, construction of the 3D model and its transmission to the client. In the client side, a PC reconstructs received 3D model and renders it. The server consists of N stereo cameras and $N + 1$ PCs. These cameras are located around a target object, and their relative positions of cameras are calculated in advance. Each camera is connected to one of the PCs through IEEE1394. The PCs are also connected to each other, and to a *"Parent PC"* which is not connected directly to any camera. The obtained 3D model is transmitted to the client from this Parent PC. The client PC reconstructs the 3D shape based on the received Point Cloud. In order to improve the image quality, we apply the cylindrical method to each frame of 3D sequence at the reconstruction stage. Details of the cylindrical method are explained in section 3. The 3D shape is reconstructed using valid points after applying the cylindrical method and rendered by point based rendering. Produced 3D model is seen by multi viewpoints of user arbitration.

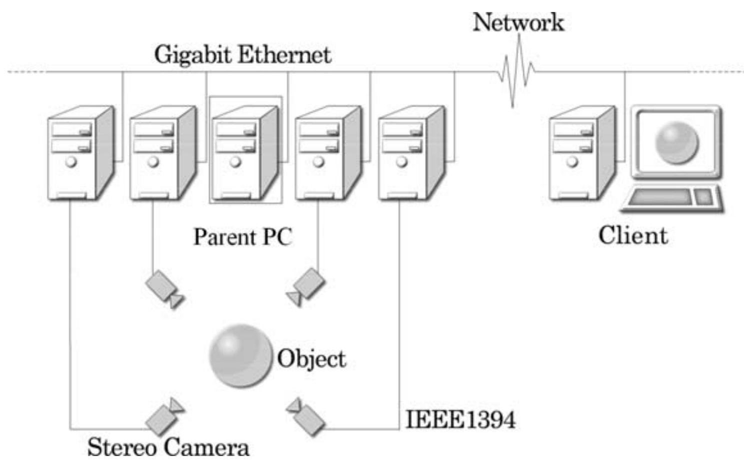


Fig. 1: Real Time 3D Imaging System: This is the client server system. The server acquires and transmits the 3D shape continuously. The client reconstructs and produces it. By repeating this procedure, it can generate the 3D sequence.

2.2 Tasks in Server

The server performs five steps. As the first step, each PC acquires color images from the connected stereo cameras. Each camera has three CCD units and can capture three color images at the same time. These color images have disparities based on the positions of the CCD units. All PCs are synchronized based on a signal sent from the Parent PC. At the second step, range images are generated using these color images by the stereo matching method. We suppose that the range images are obtained quickly using a conventional method. The third step is ROI extraction based on background subtraction. Here, the background image is taken previously without any object in front of the camera. This method removes the pixels corresponding to the background based on color and depth values of the obtained images. At the forth step, the range images taken at the same time are gathered in the Parent PC and integrated. The 3D image is constructed by integration of multiple range images. At this stage, each range image has to be rotated by certain angle with respect to a reference image. Rotation matrices for each image are calculated using relative positions of cameras in advance. Each range image is converted into world coordinates and transformed using these rotation matrices. After all range images are transformed, the 3D model is treated as a cluster of points i.e. the Point Cloud. Finally in the fifth step, the Parent PC generates packets of the Point Cloud and transmits those to the Client one by one. When the Parent PC has finished transmission, it sends a signal to other PCs to acquire the color images again from the connected camera.

2.2.1 Range Image

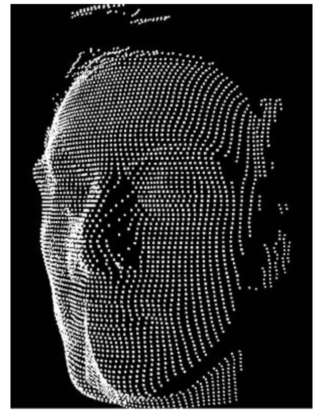
In our system, a range image is a 2D image in which each pixel has a depth value, in addition to the color information. It has depth values from only one view direction and is often called *"2.5D image"*. The depth value of each pixel is estimated by stereo matching method. The surface of the object in the 2.5D image is represented only by position of vertices, without any connectivity information.

2.2.2 Point Cloud

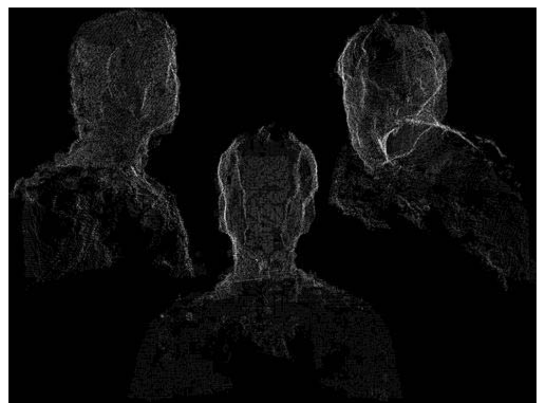
Nowadays, polygonal mesh models are used extensively as representation technique of the 3D shape. In the case of the mesh models, the connectivity information affects the data compression efficiency, because it is generally given in integer values and their coding should be lossless. Moreover, reconstructing 3D meshes by the stereo matching method generally requires the following processes: (1) Taking depth image. (2) Integration of several depth images. (3) Polygon meshing of point cloud. (4) Simplification of meshes. (5) Generating texture and mapping it to 3D mesh. The computational costs of these procedures are basically high. The feature of our method is to treat point cloud with the 3D coordinates and the color information directly without any 3D meshing. So, calculation is reduced drastically because above (3), (4), (5) processes are not required. Moreover, each depth image can be processed separately. So it can be implemented with simple parallel processing.

2.3 Tasks in Client

The client PC receives the bit-stream of the Point Cloud from the Parent PC. The received bit-stream consists of the spatial coordinates and color information of the points. The 3D shape is represented using point based geometry. At this time, overlaps and annoying artifacts are caused from failure of camera calibration or ROI extraction. In order to remove the invalid points, we apply the cylindrical method to the 3D model before rendering. The detail of the cylindrical method is explained in the section 3. The 3D shape is depicted using valid points by point based rendering. Point based rendering is one of the visualization techniques. The surface is represented by spreading each point to the colored primitives.



(a)



(b)

Fig. 2: Range Image and Point Based 3D Model: (a) range image, (b) reconstructed Point Cloud. Each pixel of range image has depth value from one directional view. Point based model is constructed by integrating multiple range images. Surfaces of range image and 3D model are consists of vertices.

2.3.1 Reconstruction

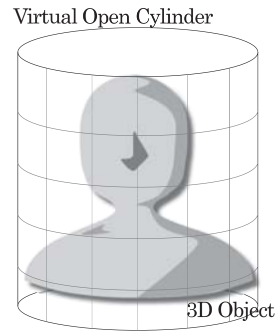
The 3D shape is reconstructed by allocating the points in the spatial space based on coordinates of received bit-streams. When all data cannot be received due to some traffic problems in network, this system is still able to reconstruct the 3D shape from the data received until then, because the point based rendering does not depend on the connectivity of the points. Even if the current frame process is interrupted due to transmission of the packets of next frame, it is still possible to handle this situation. Furthermore, unnecessary points are removed quickly and efficiently by applying the cylindrical method to each frame of 3D sequence.

2.3.2 Rendering

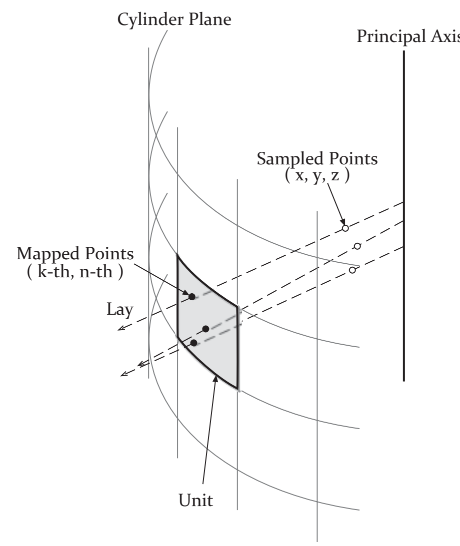
We implemented the rendering with the Point Cloud. In our system, the shape of the primitives is rectangle whose size is determined experimentally. The direction of the primitive face is decided by its normal vector. This normal vector is always facing user selected viewpoint. If the primitive's face is fixed in one direction, sideways primitives cannot be in sight because the primitives are very thin. Thus, the surface is displayed by rendering the primitives adaptively with a user selected viewpoint. The colors of the primitives are mapped based on color information of each point.

3 Cylindrical Method

In the client side, overlaps and annoying artifacts cause between different range images because the reconstructed 3D model is the integrated range images. Fig.11 (a) shows overlaps and points on the front of face in Fig.10 (a) are the annoying artifacts. By the influence of the shadow of the object itself, the annoying artifacts are left around the boundary at the ROI extraction stage. They cause image degradation and should be removed. However, since the 3D Point Cloud does not have connectivity information, it is not easy to decide the relationships among points. This makes the determination of the unnecessary points more difficult. In our previous research [10], we have proposed a voxel based method for the removal of the unnecessary points. However this approach cannot distinguish points in a same surface and ones in two closely adjacent surfaces. To address the problem, we employ a mapping method which projects the Point Cloud onto the inner wall of a cylinder. It can treat the whole Point Cloud in a single 2D plane. Note that the cylindrical method is used to determine unnecessary points, and that does not transform the coordinates of the points directly.



(a)



(b)

Fig. 3: Mapping on the Wall of the Cylinder: (a) 3D object with virtual open cylinder. (b) Image of mapping on the inner wall of the cylinder. Sampled points in 3D space are converted to mapped points in 2D plane.

3.1 Mapping

We assume that the 3D object is surrounded by a virtual open cylinder completely (see Fig.3 (a)). The axis of the cylinder is along the principal axis of the object. In practice we apply PCA to the point clouds and adopt the principal vector as the axis. The height of the cylinder is determined to cover all the points in the ROI. The cylinder and the object are cut into thin horizontal slices. Each slice gets a 2D point set from the surface of the object surrounded by a circle corresponding to the circumference of the cylinder (see Fig.4 (a)). In each slice, each point of the 2D point set is mapped on an intersection of a ray and the circle. This ray is emitted from a center of the circle and passes through that point. Therefore, a point can be represented by magnitude r and an angle ϕ based on coordinates x and z of that point (see Fig.4 (b)). The values of r and ϕ are calculated by following simple mathematical formula.

$$r = \sqrt{x^2 + z^2}, \tag{1}$$

$$\phi = \tan^{-1}\left(\frac{z}{x}\right) \tag{2}$$

The range of ϕ is 0 to 2π, where both 0 and 2π point to the same direction i.e. these are along the x axis. By repeating this process for all points in a slice, all of 2D point set is projected on the circle. Applying this procedure on all slices, entire surface of the object is mapped on the inner wall of the cylinder.

3.2 Unit Plane

After the mapping process, the cylinder's wall is cut vertically and can be warped to a plane (Fig.7). Then this plane is divided by a regular grid based on the angle ϕ and y component. These divided small domains are named as *"Units"*. Here, one of the rows corresponds to the slice. If the plane has M columns, points stored in the *k-th* Unit have angle ϕ satisfying the following condition.

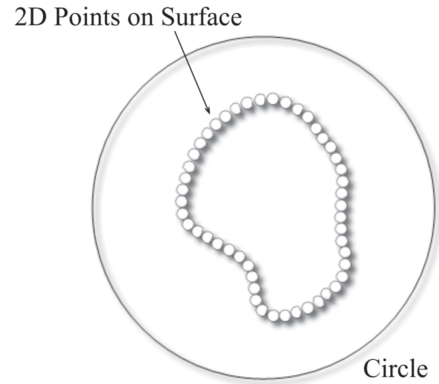
$$\frac{2\pi(k-1)}{M} \leq \phi < \frac{2\pi k}{M} \tag{3}$$

We consider the points assigned to one Unit as one segment. We can control quality and speed of processing by changing the size of the grid.

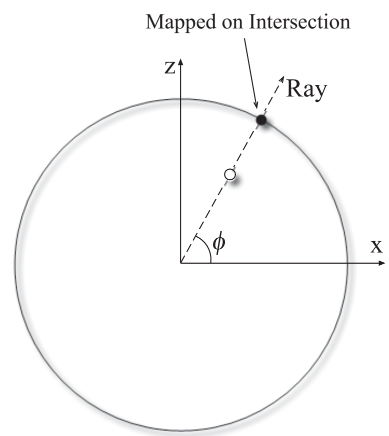
3.3 Unit Process

When there are points from more than one camera in one segment, we take it as the overlap. Fig.5 shows the image observed from positive y direction. In Fig.5 (a), one Unit stores the points from only one camera. However in (b), the points acquired from two cameras are stored and overlapped in one Unit. In this case, one of these surfaces should be removed.

First, we find a tangent vector of each surface that is composed of points from a camera. In the case of Fig.6(a), we have two vectors. These vectors are normalized. Secondly, a weighted average of the tangent vectors is computed using the points from each camera (Fig.6(b)). In other words, it is similar to computing a gravity center of each vector. By doing this, the vector estimated from many points is weighted more.



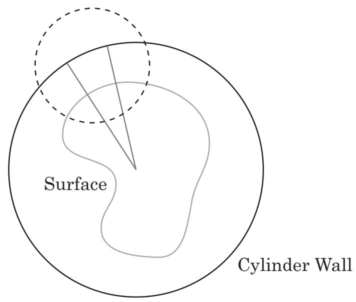
(a)



(b)

Fig. 4: Mapping based on Angle: (a) One of the horizontal slices. 2D point set are surrounded by the circle corresponding to the circumference of the cylinder. (b) Image of mapping based on angle. Each point of the 2D point set is mapped on an intersection of a ray and the circle. The ray is decided the angle ϕ calculated by x and z components of that point.

Thirdly, the inner products of the average tangent vector and view axis of each camera are calculated. We have the view axis vectors of the cameras in advance. In the case of Fig.6 (b), we have two inner products. Finally, the points which came from a camera with the minimum inner product are left, and others are removed (see Fig.6 (c)). This procedure may keep the points from the camera which lies at the direction of that surface.



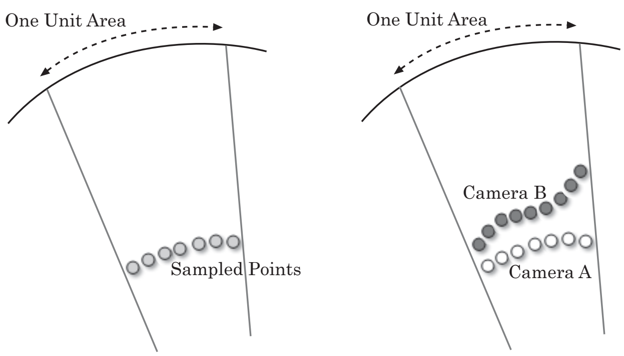


Fig. 5: Overlap in One Unit: These are images observed from y positive direction. (a) One unit stores points came from only one camera. (b) There are points came from two cameras. This time we take it as overlap. One of the surfaces should be removed in (b).

4 Experimental Results

In this research, the server consists of four PCs connected to stereo cameras and a parent PC (Pentium IV). The server equipment is shown in Fig.8. A target real object is human upper body. One of the cameras was located in front of the target and another was back. Remaining two cameras were placed at front-right and front-left. Since we set the cameras like this, it can acquire facial expression closely. The cameras were also placed at the same height because the target is similar to tubular type. This also favors cylindrical method. All of PCs were connected through Gigabit Ethernet and the client PC was located in the same LAN. We used TCP for transmitting the data between PCs. And we set resolution of each range image to 240×320. The resolution of the cylinder affects the speed and quality. After experiments on several unit sizes, we have adopted 200x73 for the cylinder resolution. In average PCs, smaller sizes may be used in a sense of computational cost. But very small sizes perform poor since one unit may contain so few points

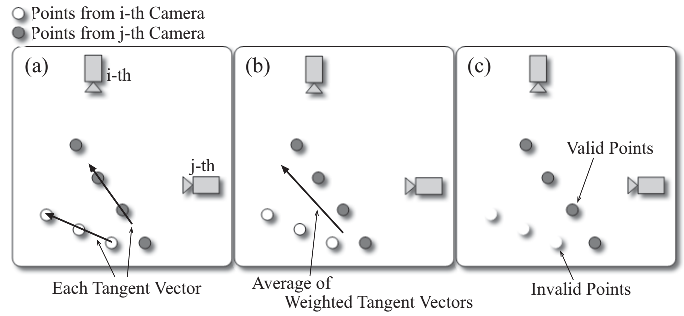


Fig. 6: Procedure of Unit Process: White points are come from *i-th* camera and gray points from *j-th*. (a) tangent vector is found for each set of points. (b) The average of weighted tangent vectors is estimated. After that, the inner products are calculated. (c) Points came from camera with minimum inner product are left.

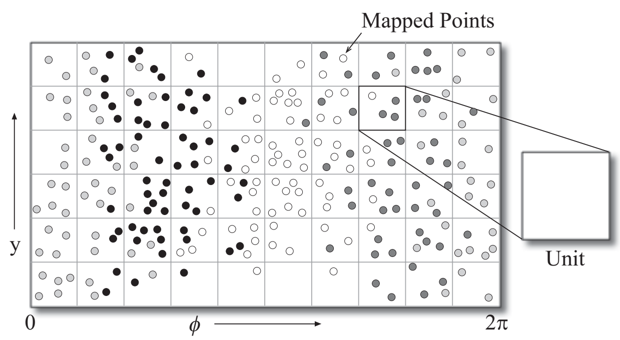


Fig. 7: Unit Plane: Unit Plane warped from the wall of the cylinder. This plane is divided by regular grid. These divided small domains are called ”Unit”. Units store mapped points.

that one cannot estimate a surface well.

At the client side, a viewer application was implemented that can represent the 3D model from multi viewpoint of user arbitration. It can zoom and rotate the 3D model arbitrarily. A sample of the viewer is shown in Fig.9. Our system generates the 3D sequences at about 5-10 frame per second (fps). The size of data actually transmitted to the client is about 2-7 Mega bit per second (Mbps). The resulting sample movie is available in [11].

Fig.10 shows the results of applying cylinder mapping to one frame of the 3D sequence. Fig.10 (a) is the 3D model reconstructed by simple integration of four range images. On the other hand, Fig.10 (b) is the model obtained by integration using the cylindrical method. Points from right and left range images are the annoying artifacts on the front of the face in (a). However in (b), these annoying artifacts are removed and the facial expression is finer than (a).

The validity of the cylindrical method is shown in Fig.11. These pictures are cut surface of its nose. Each line represents the surfaces acquired from a camera. Fig.11 (b) is the result with cylindrical method and (a) is without one. The surface which is inside the target cannot be in sight just like the dash line in (a). These overlaps were removed by the cylindrical method in (b). One frame of the 3D sequence has about 40000-45000 points. After the cylindrical method is applied, each frame has around thirty thousand points. Removal ratio was about 30%. The time to apply the cylindrical method was less than 0.1 second and it did not almost affect the sequences visually.

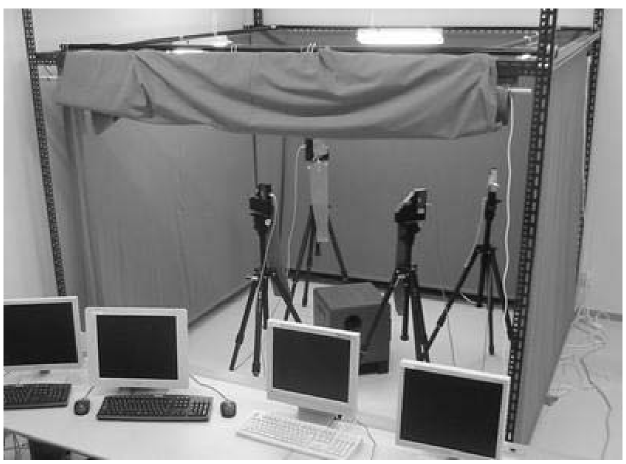


Fig. 8: Server Equipment: The server consists of four stereo cameras and five PCs. Camera and PC are connected through IEEE1394. The size of equipment is $2.5(L) \times 2.5(W) \times 2.0(H)$ meters. The distance of target and each camera is about one meter.

5 Conclusion

In this paper, we introduced a real time 3D imaging system by handling the 3D shape and the color information as Point Cloud. It could be mounting with more small scale equipment since employing the range images without voxel method. We used only 4 cameras at the modeling stage. It is an advantage of our system that it enhances the transmission efficiency by eliminating the connectivity information.

Fig. 9: Example of Client Viewer: This viewer produces 3D models continuously and translation, zoom and rotation of models are possible. It can be seen from multi viewpoint user arbitration. This is based on point based rendering using Open GL.

Moreover, by applying the cylindrical method to the real time 3D imaging system, the overlaps and the annoying artifacts are removed efficiently in real time.

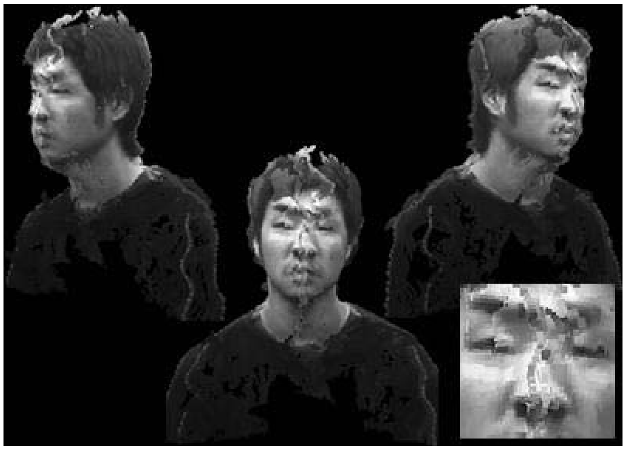
However, there still is room for improvements left. For example, image quality and frame rate should be higher, ROI extraction should be more robust and camera calibration should be more precise.

References :

[1] H. Saito, S. Baba, M. Kimura, S. Vedula and T. Kaneda, ”Appearance-based virtual view generation of temporally-varying events from multi-camera image in the 3D room,” *Second International Conference on 3-D Digital Imaging and Modeling (3DIM99)*, pp. 516–525, Oct. 1999.

[2] X. Wu, T. Matsuyama, ”Real-time active 3D shape reconstruction for 3D video,” *Proc. of 3^{rd} International Symposium on Image and Signal Processing and Analysis*, pp. 186–191, Sep. 2003.

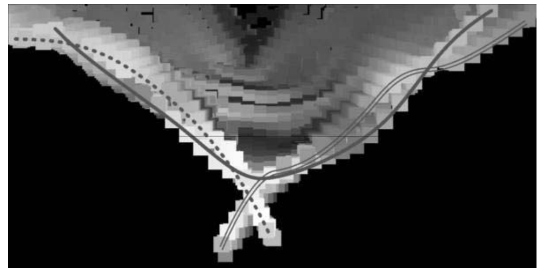
[3] M. Ueda, D. Arita, R. Taniguchi, ”Real-time free-viewpoint video generation using multiple cameras and a PC-cluster,” *Pacific-Rim Conference on Multimedia*, pp. 418–425, Dec. 2004.



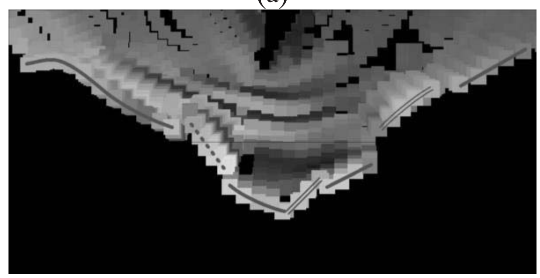
(a)

(b)

Fig. 10: Result of Cylindrical Method: These show one frame of the 3D sequence. (a) Integrated four range images. (b) After applying the cylindrical method. Annoying artifacts on the front of face are removed in (b).



(a)



(b)

Fig. 11: Horizontal Cut surface of nose: (a) without the cylindrical method. Overlapping area cannot be in sight like dash line. However overlaps are removed especially front side of face in (b).

[4] S. Würmlin, E. Lamboray, O. G. Staadt and M. H. Gross, "3D video recorder," *IEEE Computer Society Press, Proceedings of Pacific Graphics '02*, pages 325-334, 2002.

[5] S. Würmlin, E. Lamboray, M. H. Gross, "3D video fragments: dynamic point samples for real-time free-viewpoint video," *Computers & Graphics, Special Issue on Coding, Compression and Streaming Techniques for 3D and Multimedia Data*, 2004.

[6] T. Kanade, A. Yoshida, K. Oda, H. Kano and M. Tanaka, "A stereo machine for video-rate dense depth mapping and its new application," *the* 15^{th} *Computer Vision and Pattern Recognition Conference*, pp. 196–202, June. 1996.

[7] PointGrey Research, Digiclops, http://www.ptgrey.com/products/stereo.html

[8] Rusinkiewicz, S., and Levoy, M., "Qsplat: a multiresolution point rendering system for large meshes," *SIGGRAPH '00*, pp. 343–352.

[9] Isthiaq Rasool Khan, Masahiro Okuda and Shin-ichi Takahashi, "Regular 3D mesh reconstruction based on cylindrical mapping," *IEEE International Conference on Multimedia and Expo*, TP1-6-2, June. 2004.

[10] Takanori Uchida, Masaharu Kajitani and Masahiro Okuda "Point Cloud Reconstruction and Rendering for Real Time 3D Imaging," International Symposium on Information and Computer Elements ISICE Sept. 2007

[11] http://vig.is.env.kitakyu-u.ac.jp/English/research/realtime_demo.html