MODELING REALISTIC FACIAL OBJECT FROM LASER SCANNER DATA AND PHOTOGRAPHIC IMAGES FOR MALAYSIAN CRANIOFACIAL DATABASE

Deni Suwardhi, Halim Setan, Albert Chong, Zulkepli Majid, Anuar Ahmad

Medical Imaging Research Group (MIRG), Faculty of Geoinformation Science & Engineering, Universiti Teknologi Malaysia (UTM)

Tel:

Fax:

E-mail: denisuwardhi@fksg.utm.my, halim@fksg.utm.my

Abstract

This paper presents a technique to model a person's face from a set of digital photographs corresponding to different views of the 3D facial mesh from laser scanner. Modeling approach is based on photogrammetric techniques in which images are used to create precise geometry and texture information. Faces are modeled by interactively fitting a 3D facial mesh to a set of images. The fitting process consists of several basic steps. Firstly, multiple views of a human face are captured using cameras at some fixed locations. Next, a set of initial corresponding points are marked on the photographs and the face in the different views manually (typically, corners of the eyes and mouth, tip of the nose, etc.). These points are then used to automatically recover the camera parameters (position, focal length, etc.) corresponding to each photograph, as well as the 3D positions of the marked points in space. The 3D positions are then used to adjust 3D face mesh. Finally, one or more texture maps are extracted for the 3D face mesh from the photos. Either a single view-independent texture map can be extracted, or the original images can be used to perform view-dependent texture mapping.

1.0 INTRODUCTION

Creating realistic human face models is a very challenging problem, because the human face is an extremely complex geometric form. There exist different techniques to reconstruct the facial object surface and to build photorealistic 3-D models. Although the geometry can be measured by various methods of computer vision, for precise measurements laser scanners are usually used. However, most of laser scanners do not provide texture and color information, or if they do, the data is not accurate enough. Additionally, CT (Computing Tomography) is becoming readily available in many hospitals. A geometrical model derived from CT data that lacks in color is not suited for presenting simulated results to a patient.

Texture and color information could be provided by the other dataset, the collection of digital images. The task of precise fusion of the geometric and the visual data is not simple, since the pictures are taken from different viewpoints and with varying camera parameters. When the image-to-surface fusion or registration problem is solved, there is still the problem of seamless blending of multiple textures images of a surface patch appearing in different views.

This paper presents a method to add texture on the 3-D geometric person's face model by mapping texture acquired with multiple digital cameras. Based upon estimated camera

parameters (i.e. camera position and direction), a corresponding inverse projection is carried out for mapping.

2.0 MATERIAL

2.1 Laser Scanning Acquisition

In order to build a craniofacial database in Malaysia, new technologies have developed through a research program at the Medical Imaging Research Group Lab, Faculty of Geoinformation Science & Engineering, Universiti Teknologi Malaysia (UTM) that allowed researchers to capture high-resolution 3-D data points (x,y,z) of the face surface in a few seconds (Majid et al., 2005). The new scanning technology has many advantages over the old system of measurement, which uses tape measures, anthropometers (a special measuring ruler), and other similar instruments. The Minolta VIVID-910 3-D digitiser was selected and two of these scanners were used to scan the whole craniofacial area (Figure 1).

Figure 1

2.2 Digital Camera Images

Our system configured for clinical craniofacial imaging consists three sets of digital stereo cameras, which consisted of six Canon PowerShot S400 (4.0 megapixel) digital professional cameras (Figure 1). The six cameras are synchronized to capture the natural photographic appearance of the subject under normal white-light flash, just few milliseconds after the shutter switch is fired. So that the detailed geometric configuration of all of the cameras can be determined, a calibration target (comprising frame on contrasting background and of accurately known targets dimensions and location) is presented and captured by the cameras simultaneously with subject (Figure 1).

3.0 FACIAL MODELING METHOD

Figure 2 shows the methodology briefly. Firstly, pose recovery step is performed to produce geometric configuration of cameras. Secondly, 3-D geometric model transformed into photogrammetry coordinate system by using some anatomical landmarks as control points. Finally, texture information is extracted from the photos once the geometry has been estimated.

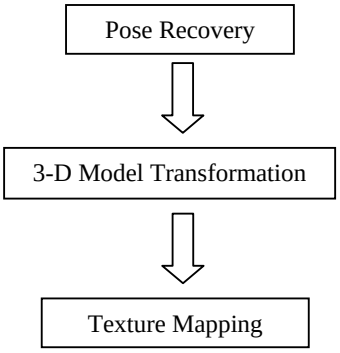

Figure 2. Facial modeling methodology

3.1 Pose Recovery

The goal of pose recovery is to estimate, for each photograph, the internal parameters (focal length, and lens distortion) and external parameters (position and orientation) of the camera. To formulate the pose recovery problem, a rotation matrix $_{k}$ and a perspective centre $^{k}_{0}$ is associated with each camera pose k. (The three rows of $_{k}$ are $\begin{bmatrix} r^k_{11} & r^k_{12} & r^k_{13} \end{bmatrix}$, $\begin{bmatrix} r^k_{21} & r^k_{22} & r^k_{23} \end{bmatrix}$ and $\begin{bmatrix} r^k_{31} & r^k_{32} & r^k_{33} \end{bmatrix}$, and the three entries in $^{k}_{0}$ are X^{k}_{0}, Y^{k}_{0}, Z^{k}_{0}) We write each 3D feature point as m_i, and its 2D image coordinates in the k-th image as (x^k_i, y^k_i).

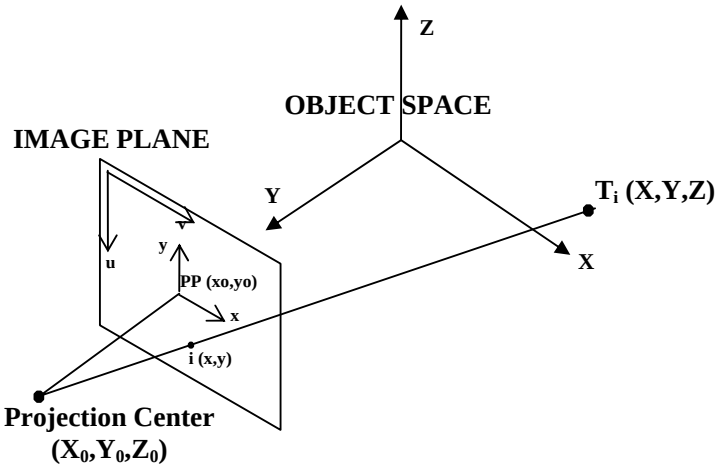

Figure 3. Collinearity Condition

A flexible and accurate photogrammetric positioning and calibration method is the bundle adjustment with selfcalibration (Atkinson 1996). The mathematical basis of the self-calibrating bundle adjustment is the collinearity model; the collinearity condition states that a point in object space, its corresponding point in an image and the projective center of the camera lie on a straight line (Figure 3). The standard form of the collinearity equations is:

$$x_i - x_0 = \frac{-c^k[r_{11}^k(X_0^k - X_i) + r_{12}^k(Y_0^k - Y_i) + r_{13}^k(Z_0^k - Z_i)]}{[r_{31}^k(X_0^k - X_i) + r_{32}^k(Y_0^k - Y_i) + r_{33}^k(Z_0^k - Z_i)]} = -c\frac{U}{W} \quad (1a)$$

$$y_i - y_0 = \frac{-c^k[r_{21}^k(X_0^k - X_i) + r_{22}^k(Y_0^k - Y_i) + r_{23}^k(Z_0^k - Z_i)]}{[r_{31}^k(X_0^k - X_i) + r_{32}^k(Y_0^k - Y_i) + r_{33}^k(Z_0^k - Z_i)]} = -c\frac{V}{W} \quad (1b)$$

All measurements performed on digital images refer to a pixel coordinate system (u,v) while collinearity equations (1) refer to the metric image coordinate system (x,y). The conversion from pixel to image coordinates is performed with an affine pixel to image coordinate system transformation. The collinearity model needs to be extended in order to meet the physical reality, by introducing some systematic errors; these errors are compensated with correction terms for the image coordinates, which are functions of a set of additional parameters (AP). A set of additional parameters widely used in photogrammetry (Atkinson 1996) consists of the parameters describing symmetrical radial lens distortion (K_1, K_2, K_3), and parameters of decentering lens distortion (P_1, P_2). The extended collinearity equations have the following form:

$$x - x_0 = -c\frac{U}{W} + \Delta x \quad (2a)$$

$$y - y_0 = -c\frac{V}{W} + \Delta y \quad (2b)$$

where

$$\Delta x = -x_0 + \bar{x}r^2K_1 + \bar{x}r^4K_2 + \bar{x}r^6K_3 + (2\bar{x}^2 + r^2)P_1 + 2P_2\bar{x}\bar{y} \quad (3a)$$

$$\Delta y = -y_0 + \bar{y}r^2K_1 + \bar{y}r^4K_2 + \bar{y}r^6K_3 + (2\bar{y}^2 + r^2)P_2 + 2P_1\bar{x}\bar{y} \quad (3b)$$

and

$\bar{x} = x - x_0$; $\bar{y} = y - y_0$; $r = \sqrt{\bar{x}^2 + \bar{y}^2}$

K_i: Parameters of symmetrical radial lens distortion

P_i: Parameters of de-centering lens distortion

Solving a self-calibrating bundle adjustment means to estimate the additional parameters in equation (3) as well as position and orientation of the camera(s) and object coordinates starting only from a set of correspondences in the images. Two collinearity equations as in (1) can be formed for each image point. Combining all equations of all points in all the images leads to a system of equations to be solved. The equations are non-linear with respect to the unknowns and, in order to solve them with a least squares method; they must be linearized, thus requiring approximations. A first order Taylor expansion is used for the linearization. Observed object coordinates are introduced via additional observation equations while geometric information in the form of additional observations (distances, angles) or geometric constraints (lines, planes) can also be used (Remondino, 1996). The additional parameters are also introduced as observations. Not all additional parameters can necessarily be determined

from a given arrangements of images and object points; moreover, non-determinable parameters (over-parameterisation) can lead to a degradation of the results. The resulting system of observation equations can be formulated in the Gauss-Markov model as (Remondino, 1996):

$$-e = A \cdot \quad -l \tag{4}$$

with:

$$^{T} = \left[\Delta X \quad \Delta Y \quad \Delta Z \quad \Delta X_0 \quad \Delta Y_0 \quad \Delta Z_0 \quad \Delta\omega \quad \Delta\phi \quad \Delta\kappa \quad AP_i\right];$$

where:

- e is the true error;
- A is the design matrix, containing the partial derivatives of the equation (1) with respect to the unknowns, evaluated with the approximations;
- is the vector of the unknowns;
- ΔX , ΔY , ΔZ are the changes to approximations of the object coordinates of a point
- ΔX_0, ΔY_0, ΔZ_0, $\Delta\omega$, $\Delta\phi$, $\Delta\kappa$ are the changes to approximations of exterior orientation elements
- AP_i additional parameters;

Calling P the weight matrix of the observations, the system is solved with the least squares solution:

$$\hat{x} = (A^T PA)^{-1} A^T PF \tag{5}$$

Due to non linear characteristic of the problem, iterations need to be performed. The residuals v of the observations and the a posteriori variance factor $\hat{\sigma}_0$ are computed as shown in (7), with r the redundancy or degree of freedom (e.g. the difference between number of equations and number of unknowns):

$$v = A \cdot \hat{x} - F \; ; \quad \hat{\sigma}_0 = \sqrt{\frac{v^T Pv}{r}} \tag{6}$$

Self-calibrating bundle adjustment is a very powerful photogrammetric calibration method. It provides for accurate orientation and location of the sensor and for accurate reconstruction of the object space. An accuracy of 1/10th of a pixel in x,y coordinates and a depth accuracy of 1/10000 of the average object distance can be obtained. As a prerequisite, the network design requires a highly convergent imaging configuration or accurate control points information otherwise the determinability of self-calibration parameters can fail and lower accuracy is reached. For more details, see (Atkinson 1996; Fraser 1997).

3.2 3-D Model Transformation

Before texture mapping is performed, 3-D model must be transformed into photogrammetric coordinate system. In such a condition the method and software is developed for generating texture on given 3-D model using images by establishing correspondence between some points of 3-D model and the image. The correspondence is established by operator, marking corresponding points (anatomical landmarks) in a 3-D model and in the given images.

The 3-D coordinates of points from the images is precisely determined by the self-calibration bundle adjustment in pose recovery step. Simultaneously, internal parameters and exterior

orientations of each camera/image are determined. The targets are used for the registration of the photogrammetrically derived coordinates with the coordinates obtained using the 3-D laser scanning technique.

The transformation includes three-dimensional rigid body translation and rotation to provide alignment and orientation and also provides metric adjustments to allow for 3-D geometric model to photogrammetry system's size differences. Scaling may be visualized as deformation, rotation and translation.

The sample of arbitrary photograph mapping on reconstructed face 3D model is shown in Figure 4. In the arbitrary image (Figure 4a) 8 reference points were marked at specific face points like a corner of the eye, and a corner of the mouth. Corresponding 8 reference points were marked in the face 3-D model.

Figure 4. 3-D geometric model transformation

3.3 Texture Mapping

In this section, we describe the process of extracting the texture maps necessary for rendering photorealistic images of a reconstructed face model from various viewpoints. The texture extraction problem can be defined as follows. Given a collection of photographs, the recovered viewing parameters, and the fitted face model, compute for each point m on the face model its texture color $T(m)$.

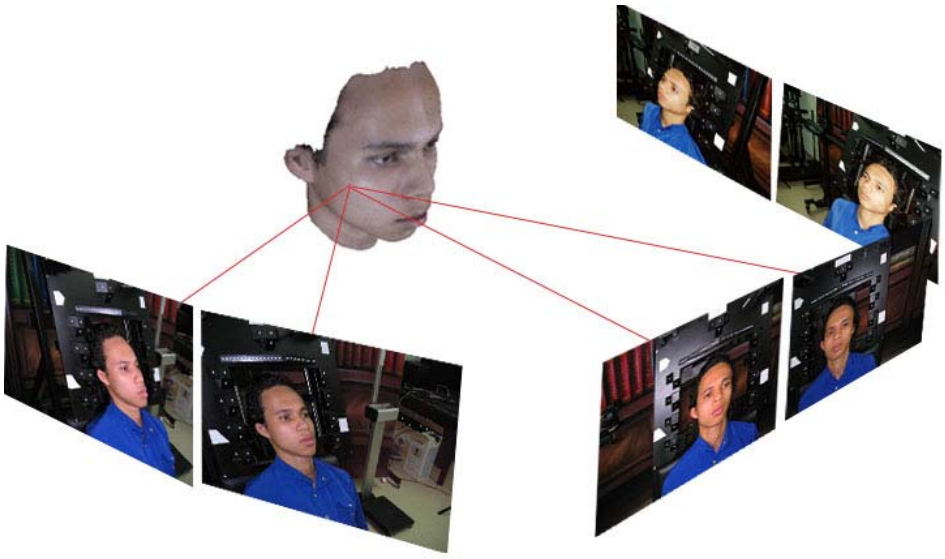

Figure 5. Sampling the photographs. The texture value at a point on the model's surface is determined by sampling the photographs where this point is visible

Each point m may be visible in one or more photographs; therefore, we must identify the corresponding point in each photograph and decide how these potentially different values should be combined (blended) together. The photographs are sampled by projecting m using the recovered camera parameters. Figure 5 illustrates this process. There are two principal ways to blend values from different photographs: view-independent blending, resulting in a texture map that can be used to render the face from any viewpoint; and view-dependent blending, which adjusts the blending weights at each point based on the direction of the current viewpoint (Debevec et al. 1996). Rendering takes longer with view-dependent blending, but the resulting image is of slightly higher quality (see experimental result).

3.3.1 Color matching

Different view angles and specular highlights may result in color discrepancies between photographs taken simultaneously. For better color consistency in face textures extracted from photographs, color correction or color matching should be applied to simultaneous photographs of each view. Since photographs that correspond to different views are smoothly blended to generate a texture, color discrepancies between them appear as natural lighting variations as the viewing direction changes. However color discrepancies between photographs corresponding to different view introduce unnatural skin tone variations.

In this work, we used histogram matching method to match colors between photographs. Histogram matching is the process of automatically modifying the transform line(s) for one or more datasets to force their output histograms to match the output histogram of a reference dataset. This is a standard technique used to match color across a mosaic of datasets to help minimize seams and make them appear to be one continuous image. Figure 6.a. and 6.b. shows the 'mosaic' of images on 3-D model before and after color matching, respectively.

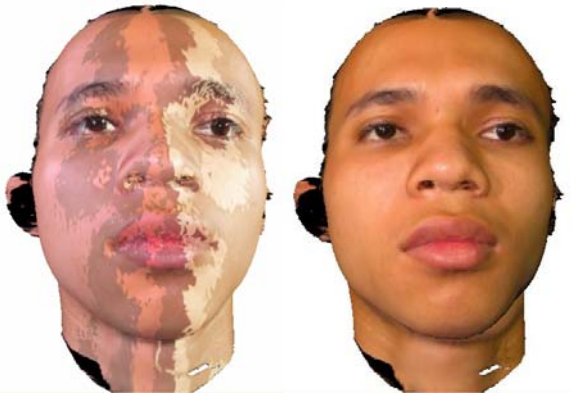

Figure 6 (a) before and (b) after color matching

3.3.2 Weight maps

As outlined above, the texture value $T(m)$ at each point on the face model can be expressed as a convex combination of the corresponding colors in the photographs:

$$T(m) = \frac{\sum_k g^k(m) C^k(x^k, y^k)}{\sum_k g^k(m)} \tag{7}$$

Here, C^k is the image function (color at each pixel of the k-th photograph,) and (x^k, y^k) are the image coordinates of the projection of m onto the k-th image plane.

The weight map $g^k(m)$ is a function that specifies the contribution of the k-th photograph to the texture at each face surface point. The function could be notated as follows:

$$g^k(m) = f(g_o^k(m), g_\varpi^k(m), g_\psi^k) \quad (8)$$

There are several important considerations that must be taken into account when defining a weight map (Pighin 1999):

1. Self-occlusion: $g^k(m)$ should be zero unless m is front-facing with reverence to the k-th image and visible in it. The first weight term $g_o^k(m)$ represents a binary (0 or 1) visibility map for each cylindrical texture map to handling self-occlusion.
2. Smoothness: the weight map should vary smoothly, in order to ensure a seamless blend between different input images.
3. Positional certainty: $g^k(m)$ should depend on the "positional certainty" [43] of m with respect to the k-th image. The positional certainty is defined as the dot product between the surface normal at m and the k-th direction of projection, $g_\varpi^k(m) = \overrightarrow{n_m} \cdot \overrightarrow{d_m^k}$ where $\overrightarrow{n_m}$ is the surface normal at facial point m whose cylindrical projection is (u,v), and $\overrightarrow{d_m^k}$ is a unit vector pointing from m to the k-th direction of projection.
4. View similarity: for view-dependent texture mapping, the weight $g^k(m)$ should also depend on the angle between the direction of projection of m onto the j-th image and its direction of projection in the new view.

3.3.3 View-independent texture mapping

In order to support rapid display of the textured face model from any viewpoint, it is desirable to blend the individual photographs together into a single texture map. This texture map is constructed on a virtual cylinder enclosing the face model. The mapping between the 3D coordinates on the face mesh and the 2D texture space is defined using a cylindrical projection, as in (Pighin 1999).

For view-independent texture mapping, we indexed the weight map g^k by the (u,v) coordinates of the texture being created. Each weight $g^k(u,v)$ is determined by the following steps:

1. Construct a visibility map $g_o^k(u,v)$ for each image k. These maps are defined in the same cylindrical coordinates as the texture map. We initially set $g_o^k(u,v)$ to 1 if the corresponding face point m is visible in the k-th image, and to 0 otherwise. The result is a binary visibility map, which is then smoothly sampled from 1 to 0 in the area of the boundaries. A cubic polynomial is used as the sampling function.
2. Compute the 3-D point m on the surface of the face mesh whose cylindrical projection is (u,v) (see Figure 6). This computation is performed by casting a ray from (u,v) on the cylinder towards the cylinder's axis. The first intersection between this ray and the face

mesh is the point m. Let $g_{\varpi}^{k}(m)$ be the positional certainty of m with respect to the k-th image.

3. Set weight $g^{k}(u,v)$ to the product $g_{o}^{k}(u,v) \cdot g_{\varpi}^{k}(m)$

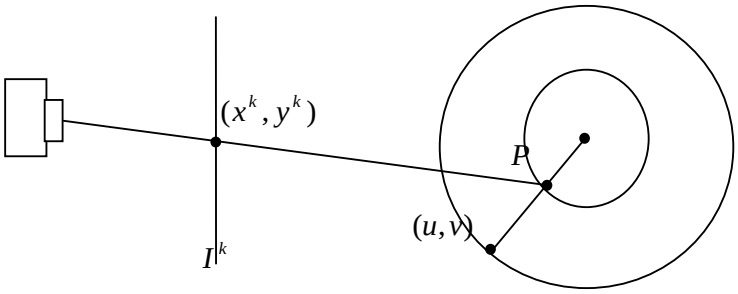

Figure 7. Geometry for texture extraction

For view-independent texture mapping, we will compute each pixel of the resulting texture T(u; v) as a weighted sum of the original image functions, indexed by (u; v).

Figure 8 illustrates the extraction of a cylindrical texture map from the photographs in Figure 5.

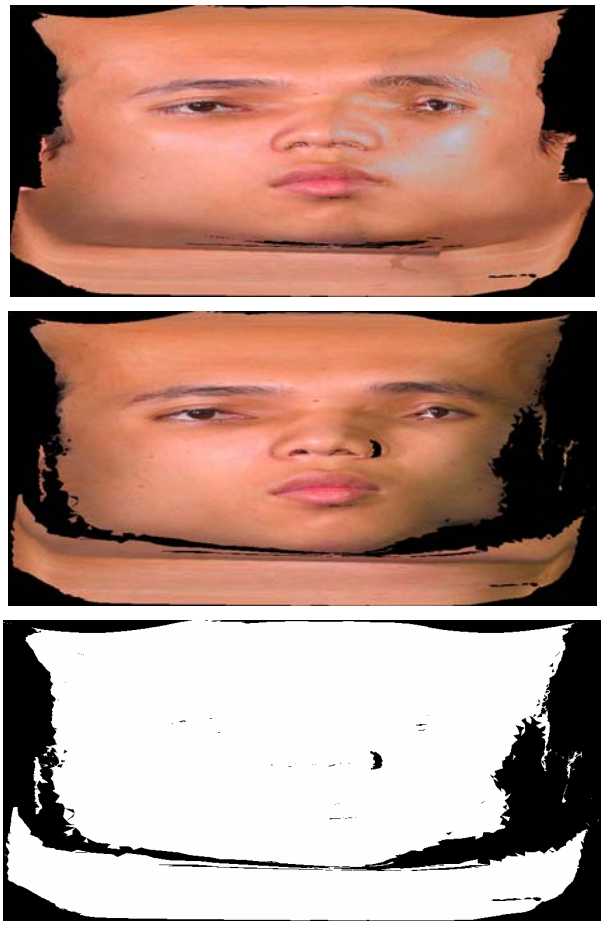

Figure 8. Example of cylindrical texture. A cylindrical texture (part (a)) has been extracted from the images shown in Figure 5. Part (b) and part (c) shows the projection of the front view and its associated weight map respectively.

3.3.4 View-dependent texture mapping

The main disadvantage of the view-independent cylindrical texture map described above is that its construction involves blending together re-sampled versions of the original images of the face. Because of this re-sampling, and also because of registration errors, the resulting texture is little blurry. This problem can be eliminated using a view-dependent texture map (Debevec et al. 1996) in which the blending weights are adjusted dynamically, according to the current view.

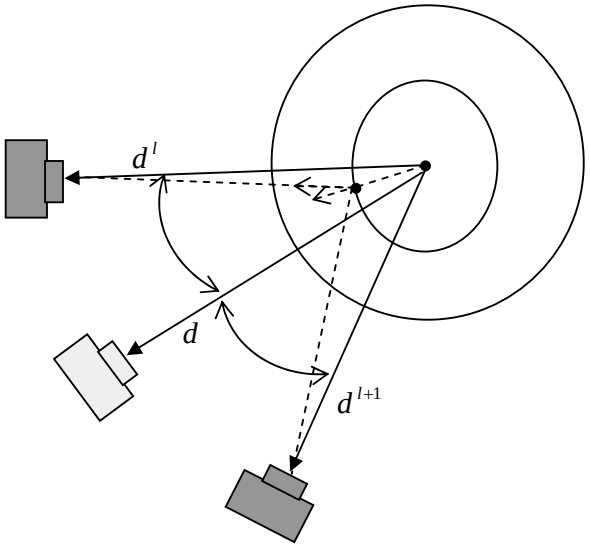

Figure 9. Example of cylindrical texture.

For view-dependent texture mapping, the model is rendered several times, each time using a different input photograph as a texture map, and blend the results. More specifically, for each input photograph, texture coordinates and a blending weight are associated with each vertex in the face mesh. Given a viewing direction d, first the subset of photographs used for the rendering are selected and then assign blending weights to each of these photographs. Since the cameras were positioned roughly in the same plane, we select just the two photographs whose view directions d^l and d^{l+1} are the closest to d and blend between the two.

In choosing the view-dependent term $g_\psi^k(d)$ of the blending weights, we wish to use just a single photo if that photo's view direction matches the current view direction precisely, and to blend smoothly between the nearest two photos otherwise. We used the simplest possible blending function:

$$g_\psi^k(d) = \begin{cases} d.d^k - d^l.d^{l+1} & \text{if } l \le k \le l+1 \\ 0 & \text{otherwise} \end{cases}$$

For the final blending weights $g^k(m)$, we then use the product of all three terms $g_o^k(u,v) \cdot g_\varpi^k(m).g_\psi^k(d)$

View-dependent texture maps have several advantages over cylindrical texture maps. First, they can make up for some lack of detail in the model. Second, whenever the model projects

onto a cylinder with overlap, a cylindrical texture map will not contain data for some parts of the model. This problem does not arise with view-dependent texture maps if the geometry of the mesh matches the photograph properly.

One disadvantage of the view-dependent approach is its higher memory requirements and slower speed due to the multi-pass rendering. Another drawback is that the resulting images are much more sensitive to any variations in exposure or lighting conditions in the original photographs.

4.0 EXPERIMENTAL RESULT

In order to test our technique, we photographed a patient (ID B252591) from University Science Malaysia (USM) Hospital in Kota Bharu, Malaysia. The photography was performed using six cameras simultaneously. The cameras were calibrated using Australis (camera calibration software developed at the University of Melbourne, Australia) and self-calibration technique was used. The two lenses had different focal lengths from other four lenses. Since no special attempt was made to illuminate the subject uniformly, the resulting photographs exhibited considerable variation in both hue and brightness. Six typical images are shown in Figure 10 and the configuration of the cameras are shown in Figure 11.

Figures 12 show the wire-frame of 3-D geometric model from laser scanner, 3-D geometric model draped with original texture image from laser scanner and 3-D geometric model draped with texture image from photographs by using our technique. The view-independent rendering is shown on the Figure 13 top and the view-dependent rendering on the bottom. Higher frequency details are visible in the view-dependent rendering.

Figure 10. Input Photographs.

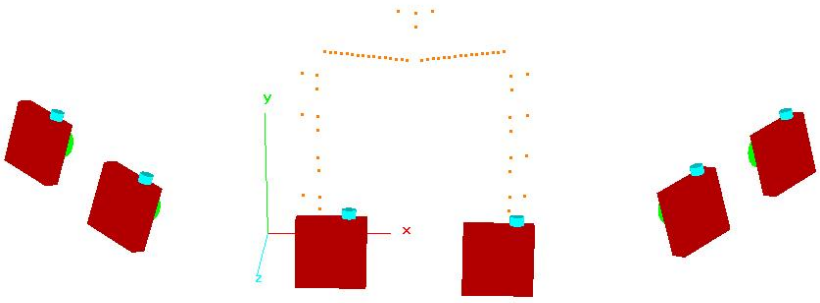

Figure 11. Camera configuration and target on calibration frame

Figure 12. Wireframe of facial soft-tissue from (a) CT, and (b) laser scanner

Figure 13. Comparison between view-independent (top) and view-dependent texture mapping (bottom).

5.0 CONCLUSION

In this paper, we have presented a modeling technique to generate realistic texture-mapped 3-dimensional models from photographs. Our technique consists of transforming a 3-D face model to a set of photographs. The model is transformed by manually specifying a set of correspondence points. We presented an accurate calibration technique to recover the camera parameters for each photograph. Finally, we discussed the extraction of view-independent and view-dependent texture maps.

There are two main technical contributions. The first is to increase the quality of cameras geometry configuration (interior and exterior parameter) by using calibration frame surround the subject. Usually, other works only use anatomical landmarks as controls for the determination of interior and exterior parameter. That way is not guarantee the quality of the geometry configuration. The second is to increase texture image resolution and to add texture image for 3-D geometric model from laser scanner and CT scanner, subsequently. 3-D laser scanners usually only provide fairly limited texture information. These advantages come at the expense of user intervention. The need for user intervention cannot be avoided until reliable automatic techniques are developed.

The future work is to construct a fully automated modeling system, which would automatically find features and correspondences with minimal user intervention. This is a challenging problem indeed, but recent results on 2-D face modeling in computer vision give us reason for optimism.

REFERENCES

1. Atkinson K.B. 1996. *Close Range Photogrammetry and Machine Vision*. Whittles Publishing, Scotland, UK.
2. Pighin F. 1999. *Modeling and Animating Realistic Faces from Images.* PhD Thesis. University of Washington, USA.
3. Alvin W. K. Soh, Yu Zhang, Edmond C. Prakash, Tony K. Y. Chan and Eric Sung. 2002. *Texture mapping of 3D human face for virtual reality environments*. International Journal of Information Technology. Vol. 8, No. 2, pp. 54-65.
4. Fraser C.S. 1997. *Digital Camera Self-calibration*. ISPRS Journal of Photogrammetry and Remote Sensing. Vol. 52, pp. 149-159.
5. Debevec P.E., Taylor C.J. and J. Malik. 1996. *Modeling and rendering architecture from photographs: A hybrid geometry- and image-based approach*. In SIG- GRAPH 96 Conference Proceedings. pp 11-20. ACM SIGGRAPH.
6. Remondino, F.: 3-D *reconstruction of articulated objects from uncalibrated images.* 3Dimensional Image Capture and Applications V, SPIE Proc., Vol. 4661 (2002) 148-154