Development of a 3D Printable Facial Model Construction System

Soonchul Jung, Yoon-Seok Choi, Jin-seo Kim
Electronics and Telecommunications Research Institute
{s.jung, ys-choi, kjseo}@etri.re.kr

Abstract

In this paper, we present a novel system that can construct the 3D printable facial models in a short time. To construct such 3D models, the system has a pipeline that consists of six steps – image acquisition, camera calibration, background separation, stereo matching, reconstruction of a facial surface, and adaptation of a reference model. We describe each step in the pipeline in detail. In the stereo matching step, the most important step among all, we use a hierarchical block matching which is favorable to a human face. Our system provides a unified component-based software which handles all the steps in one place in a short time.

Keywords: Face, Stereo Matching, Adaptation

1. Introduction

Thanks to the popularity of inexpensive 3D printers, the demands for customized figures with the individuals' faces have been increased in recent years. Such demands have led to a significant interest in 3D facial reconstruction. In order to make such a figure, we should not only reconstruct a 3D facial surface, but also create a 3D printable model from that by adapting a 3D printable reference model to the reconstructed facial surface.

A lot of steps are required to complete the work which is described in Figure 1. Each step is quite independent of each other so that it is easy enough to develop functions of one step as an independent component to others.

The stereo matching step is especially important to achieve better quality models [1]. Stereo matching method can be divided into feature-based and area-based matchings respectively. Area-based matching algorithm is often used in reconstructing facial surfaces because the face images have wide non-textured areas from which it would be difficult to detect features.

In this paper, we present a novel system that can construct 3D printable facial models in a short time. It consists of a face-capturing hardware and its software for creating 3D printable facial models from several images. The hardware has an outlook of a box attaching multiple DSLR (Digital Single Lens Reflex) cameras and light sources. The software is based on components so that operators can handle all the steps in one place. We also describe each step in the pipeline, including the facial surface reconstruction step in this paper.

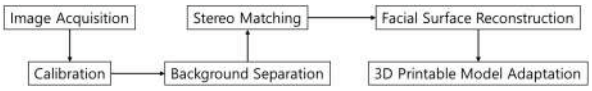

Figure 1: Steps required for 3D printable facial model construction

2. Hardware

Figure 2 shows our 3D facial scanner. It has been improved from [2] in that it has more up-to-date cameras (Canon 100D) and a more convenient chin support mechanism.

Figure 2: Our facial scanner

3. Proposed Facial Modeling Method

Our software was developed based on QT, VTK, and OpenCV open source libraries. It consists of six individual components including image acquisition, camera calibration, background separation, stereo matching, reconstruction, and adaptation components. The details are presented as follows.

Figure 3: Live view display

3.1 Image acquisition

Our system uses three Canon 100D cameras and the camera SDK (Software Development Kit) provides live view APIs (Application Programming Interfaces) as a supporting tool for image capture. Figure 3 shows an example of a live view display. This function helps us to take a good picture when the subject comes to a nice position. The person is asked to wear a hat in order to cover his face wider. Figure 4 shows a set of captured images.

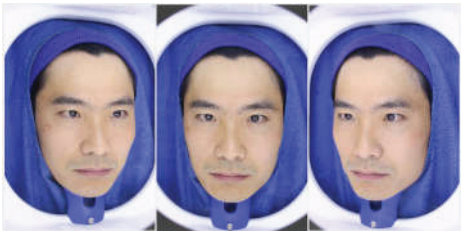

Figure 4: Captured images; Left, middle and right cameras

3.2 Camera Calibration

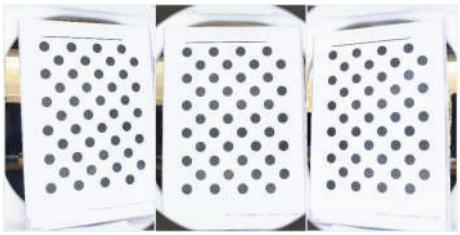

Figure 5: Example of pattern images

OpenCV (Open Source Computer Vision) is a library for the programming functions mainly aimed at real-time computer vision. It is known that asymmetric circle patterns are better than simple chessboard patterns. We captured about 10 sets of asymmetric circle pattern images given in Figure 5. We utilized the OpenCV calibration function to estimate the intrinsic and extrinsic matrices of cameras.

It takes about 10 minutes to move the pattern board, capture the images, and perform the camera calibration. Fortunately, we only have to perform this step only once during the system setup.

3.3 Background separation

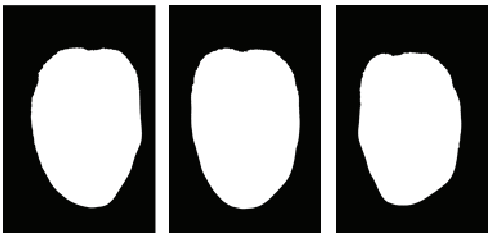

Figure 6: Each background mask

We separate a region of a human face from that of the background by Grab Cut method [3]. Skin colors of human faces are known to usually be in the range of (30, 133, 77) <= (Y, Cr, Cb) <= (230, 173, 127). We assume that the boundaries of captured images belong to the background region. We perform grab cut operation based on these information to get an initial background binary mask. After that, we find the largest connected component, clear other foreground components, and fill holes within the largest component through morphological operations. Figure 6 shows a set of the resulting masks corresponding to the images of Figure 4.

3.4 Stereo matching

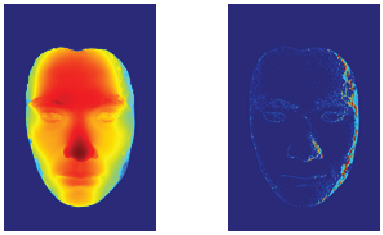

Figure 7: Resulting disparity map and constraint map

We utilize a kind of area-based block matching algorithm for stereo matching due to its simplicity and short computation time. We use NCC (Normalized Cross Correlation) as a matching cost function. NCC is known to be more robust for illumination changes [4]. Matching is done pairwise between neighboring cameras --- (left, middle) and (middle, right). Because the size of captured images is very big, an image pyramid is constructed. The image resolution at the lowest layer is chosen to be lower than 200 pixels of the width.

Our stereo matching algorithm repeats the following procedure in each layer.

Firstly, matches are computed for all pixels by using block matching in each layer. We utilize the disparity estimates from the preceding layer to limit search ranges.

Secondly, we smooth the disparity map by use of bilateral filter since the filter is advantageous for the implicit smoothness of human faces.

Thirdly, we check three constraints – smoothness, ordering and uniqueness – for all pixels. Consider a pixel p in the reference image, and the corresponding pixel q in the other image. Smoothness constraint means that the disparity of pixel p should be consistent with neighbors in a surrounding window. In this paper, pixel p satisfies this constraint if the disparity difference between p and its neighbor is less than one pixel for at least its four neighbors. Ordering constraint means that the disparity of p

should not be larger than that of its right neighbor by more than one. Uniqueness constraint means if p in the reference image matches to q in the other image, the q should also match to pixel p. In this paper, p does not violate the uniqueness constraint if $|p-q| \leq 1$.

Lastly, the disparities from violated pixels are recomputed utilizing those of neighbors.

Figure 7 shows the resulting disparity map and constraint map for the left and middle images. The disparity map looks good in appearance. In the constraint map, the foreground pixels having the same color as the background have not been violated. The pixels of green, red, or light blue colors are violated ones.

In fact, there is no ground truth for the disparity maps of human face images captured by our system, which makes quality evaluation difficult. As an alternative evaluation index, we calculate unviolated pixel ratio (# of unviolated pixels / # of face pixels). The constraint map in Figure 7 shows that most pixels in the face region are unviolated. In this case, the unviolated pixel ratio was about 0.80. The violated pixels mainly crowded the right side of the face outline and the nose. The regions are not observed in the left image, and so a lot of disparity estimates in the region are wrong.

3.5 Facial Surface Reconstruction

In order to reconstruct the facial surface, we generate 3D points from the disparity maps and triangulate them.

We obtain two disparity maps from three captured images. We re-project only unviolated pixels from the disparity maps to 3D points. A lot of 3D points are overlapped because each pixel in the middle image is re-projected twice from the two disparity maps. Therefore, we quantize the 3D points to the predefined grain size.

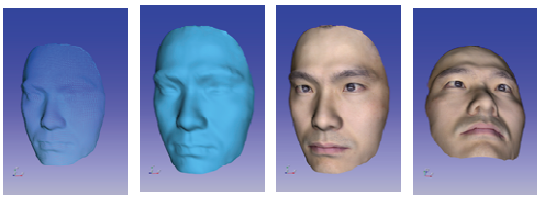

Figure 8: Result of 3D reconstruction

We use Delaunay triangulation to create meshes from the 3D points. Figure 8 shows the resulting facial surface reconstructed from the captured images. The surface model looks good but is not water-tight meaning that we cannot 3D print the model at this stage.

3.6 Adaptation

We prepared a bust model shown in Figure 9(a) as a 3D printable reference model. We interactively move, rotate and resize the reference model to match the facial surface like Figure 9(b). And then the points of the reference model near the facial surface are moved toward the facial surface in order to complete the final 3D printable model shown in Figure 9(c). This model is water-tight, and we can 3D print the model.

Figure 9: Result of adaptation to facial surface

We printed the final model to our FDM based 3D printer. Figure 10 shows the real 3D printed object.

Figure 10: 3D-printed real object

4. Conclusions

In this paper, we presented a novel system that can construct the 3D printable facial models in a short time. Our system enables an operator to handle from face scanning to 3D printable model generation in one place.

It took about ten minutes to perform all the steps except for the camera calibration. It is not important to time how long it takes to perform the camera calibration because it is performed only once during the system setup.

Utilizing our system, one of potential business models is the on-site personalized 3D figure business which offers the famous character figures applied with a buyer's face within one hour.

Acknowledgement

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 2014.

References

[1] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, and M. Gross, “High-quality passive facial performance capture using anchor frames”, *ACM Transactions on Graphics*, vol. 30, pp. 75:1-75:10, 2011.

[2] S. Kwon, S. Lee, J. W. Kim, and J. S. Kim, “3D Facial Scanner using True Colour Reproduction Technology”, *International Colour Association, 2013*.

[3] C. Rother, K. Vladimir, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts", *ACM Transactions on Graphics*, 2004.

[4] P. Leclercq, J. Liu, A. Woodward, and P. Delmas, “Which stereo matching algorithm for accurate 3D face creation?”, *Combinatorial Image Analysis*, pp.690-704, 2005.