3D Reconstruction of a Human Face with Monocular Camera Based on Head Movement

Ben Yip and Jesse S. Jin

School of Information Technologies
The University of Sydney
Sydney, NSW 2006, Australia

{benyip; jesse}@it.usyd.edu.au

Abstract

Constructing three-dimensional model from two-dimensional images is an old problem in the area of computer vision. There are many publications and our approach is specifically designed for constructing the depth map of a human face, based on the head movement in a monocular setting. In our example, along with the front view image of the user, three additional images with various head movement are also captured. The objective of our algorithm is to construct the depth map of the front view image. The head pose of the images facing left, up and right are calculated with reference to the front image. The depth map is calculated through a triangular mesh. The nodes on the mesh are the feature points that we calculate the depth with. Through image registration process, the feature points on the front view image are mapped to the other three images. Based on the head pose and the newly mapped coordinate, we could calculate the depth of the feature point. The depth results calculated from each of the three images are combined together to find the final depth value. In this paper, we assumed that the only movement in the scene is the head movement. The result is not as accurate as we expect, and we believe it could be improved.

Keywords: 3D reconstruction, monocular, head movement

1 Introduction

Estimating the shape of an object in the real three dimensional world utilizing one or more two-dimensional images, is a fundamental question in the area of computer vision.

The depth perception of a scene or an object is known to human mostly because the vision obtained by each of our eyes simultaneously, could be combined and formed the perception of a distance. However, in some specific situations, human could have a depth perception of a scene or an object with one eye when there is other additional information, such as lighting, shading, interposition, pattern or relative size. This is why it is possible to estimate the depth of a scene or an object with a monocular camera.

There are many researches in the area of depth estimation with monocular vision. Photometric stereo obtains the depth by varying the light source (Georghiades, 2003). It is possible to estimate the depth of an object based on a known movement of the object. The usage of a turning table is a good example (Lyness, et al. 2001). Brand and Bhotika, 2001, constructs the depth by using mathematic modelling of different video motion. Ziegler et al, 2001, constructs the 3D scene by first labelling the region. All depth estimations with monocular vision must take advantage of additional information about the environment, the object, or the movements.

The approach in this paper takes the additional information that the object of interest is a human head, and the human head has different pose.

1.1 Head movement

The rotation of the head involves the cervical vertebra. Biologically, there is no such point as the pivot of rotation. But for the purpose of modelling the head movement, we decided to take the vertebra at C3 as the pivot point of rotation.

It is important to notice that finding the pivot point from the front view of the user depends on the degree tilted of the face. For a straight view, C3 is roughly located at the chin area, and for a face viewing 20˚ downward, it is roughly at the area between the nose and the mouth. This is depicted in Figure 1. The position of the pivot point is better estimate from the height and the depth of the head, which is roughly at 0.8 of its height from the top and 0.7 of its depth from the front.

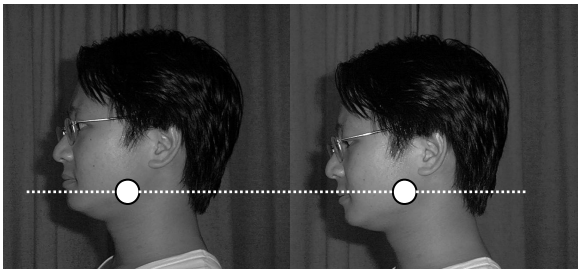


Figure 1. The white circles depict the modelled pivot point of rotation, and its location should not be estimated from the front view.

1.2 Outline of our approach

We take four images of the person when facing straight, left, up and right as shown in Figure 2. Each image has 24-bit RGB colour with resolution of 392 x 440 pixels.

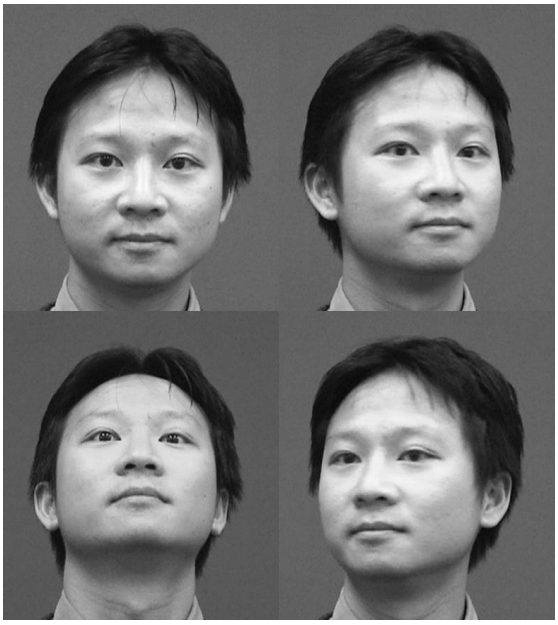


Figure 2. Our algorithm constructs the depth map for the front view image (top left). The other three images are used to help determine the depth.

The objective of our algorithm is to construct the depth map of the front view image. The head pose of the images facing left, up and right are calculated with reference to the front image. The depth map shall be calculated through a triangular mesh. The nodes on the mesh are the feature points that we calculate the depth with. Through image registration process, the feature points on the front view image are mapped to the other three images. Based on the head pose and the newly mapped coordinate, we could calculate the depth of the feature point. The depth results calculated from each of the three images are combined together to find the final depth value. A high-level pseudo code is depicted in Figure 3.

```
FV = ObtainFrontView()
FP = LocateFeaturePoint(FV)
TM = CreateTriangularMesh(FP)
For each I = ObtainAdditionalImage()
   Angle = FindPoseOfImage(FV, I)
   (Map,Similar)=ImageRegis(Angle, FP, FV, I)
   CalculateDepth(Map, FP, I)
End For
CombineCalculateDepth(FP,Similar)
DisplayDepthMap(TM)
```

Figure 3. A high-level pseudo code for the algorithm

2 Pose determination of the head

In the process of 3D reconstruction, we need to know the pose of the additional images, and the pose determination algorithm is run for each image.

There are many researches on pose determination of human head. Some use a feature of the face to determine the pose. Zitnick et al, 1999, suggested tracking the head orientation by nostril in the appendix of their paper. Ji and Yang, 2002, determined the head pose by the pupils' size, inter-pupil distance, and pupils shape. There are other approaches that use several feature points. Gemmell et al, 2000, determined head pose by performing gradient descent of nine feature points along with the face model. Ho and Huang, 1998, used the corners of eyes and mouths to determine the pose of the face in a monocular camera. Gee and Cipolla, 1994, determined the head pose by assuming various ratios of the human face between the eye corners, mouth corners and the nose tip. All publications above have some assumptions of the properties of the human face, or have a predefined face model.

We find the head pose based on the eye displacement of the middle point of the eyes to the pivot point of the head. The pitch, yaw and roll angle could be found by comparing the eye displacement from an image with the unknown pose, to the eye displacement of the known pose.

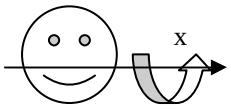
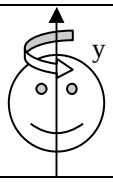
		-ve	0˚	+ve
α - angle on YZ plane from Y to Z clockwise (pitch)				
β - angle on XZ plane from X to Z anti-clockwise (yaw)				
γ - angle on XY plane from X to Y anti-clockwise (roll)				

Figure 4. The centre of eyes displacement from the pivot point of the head varies in different head pose.

Figure 4 depicts the relative changes of the eye location with various head pose. The arrows represent the displacement from the pivot point of the head to the centre of the eyes. This displacement has larger Y value if the head is tilted upwards and less or even negative if tilted downwards, as depicted in the first row of Figure 4. The eye displacement shears to our left (or right) indicates the user is turning the head to his/her right (or left). This is shown in the second row of Figure 4. The eye displacement is rotated, if the head is rolled, as depicted in the last row of Figure 4. Based on the head movement from the known pose, the unknown pose could be easily calculated. The details of actual calculation could be seen in Yip, 2004.

In our example, the image facing the left has pitch of 5.188˚, yaw of 14.36˚ and roll angle of -0.396˚. The image facing upwards has pitch of 25.72˚, yaw of 1.069˚ and roll angle of -0.415˚. The image facing right has pitch of 5.823˚, yaw of -22.27˚ and roll of -3.443˚.

3 Feature point and triangular mesh

There are many ways to determine a feature in a given image. A feature point could be obtained from edge detection. It could be the mean position of a region, or a cluster of points. In our approach, a regular triangular mesh is used because it guarantees an even distribution of the feature points. Figure 5 depicts a sample of regular triangular mesh with 49 (7 x 7) feature points. In the examples of this paper, we used 3 different mesh dimensions: 33 x 37, 49 x 55 and 98 x 110.

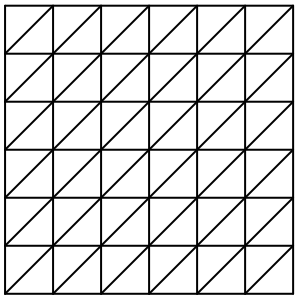


Figure 5. A sample of regular triangular mesh.

4 Image registration

Once the feature points of the front view image are defined, we are interested to know where the feature points correspond to in the three additional images. It is possible to apply feature tracking algorithm in our application (Krishnan and Raviv, 1995).

In our approach, we use image registration instead. As suggested in Brown, 1992, the registration methods can be viewed as different combinations of choices for a feature space, a search space, a search strategy, and a similarity metric. This paper uses a very simple and less computational time approach.

4.1 Feature space and search space

The feature space used in this algorithm is an 11 x 11 block of RGB colour, with the feature point in the middle. The search space used in this algorithm is a 31 x 31 block of RGB colour, with the expected feature point in the middle of the point.

4.2 Search strategy

It is important to find a good expected position. From the pose determination algorithm, we know the pose angle of the image. We could then estimate the expected position of each of the feature point by applying a rotation of the feature point of the same angle. However, calculating the rotation requires the z value of the feature point, which of course is the objective of the whole exercise. Commonly, iterations are used in this type of self recursion situation. In our approach, we estimate the z value by modelling the front view of the human head as an ellipsoid. The details of ellipsoid modelling could be found in Yip, 2003.

4.3 Similarity metric

The similarity basically is defined as the average colour difference for each of the pixel in the feature space. Let F_{ij} be the pixel at the (i, j) position of the feature block, with (0,0) equals the feature point fp. Let S_{ij} be the pixel at the (i, j) position of the search block with (0, 0) equals the searching point sp. Let $R(p_1, p_2)$ denotes the absolute difference of the red channel for pixel p_1 and pixel p_2. $G(p_1, p_2)$ and $B(p_1, p_2)$ are similarly defined for colour green and blue. The similarity is defined as:

$$s = \lambda \sum_i \sum_j \frac{R(F_{ij}, S_{ij}) + G(F_{ij}, S_{ij}) + B(F_{ij}, S_{ij})}{3.\|F\|}$$

where $\|F\|$ is the size of the feature space, which is 121 in our example. λ is a scalar, ranges between 1 and 3, to panelise points further away from the expected position. It is based on the block distance from the search position to the feature point position.

$$\lambda = (1 + \frac{|sp.x - fp.x|}{\max X} + \frac{|sp.y - fp.y|}{\max Y})$$

(maxX, maxY) is the furthest point away from the feature point, which is (15, 15) in our example.

Other than the matching position is found, the process of image registration also outputs the similarity values, which is used in the combined depth calculations later on.

5 Depth calculation for each feature point

The depth value can be calculated by the changes of pitch and the yaw angle of the pose, along with the matching position from the image registration process.

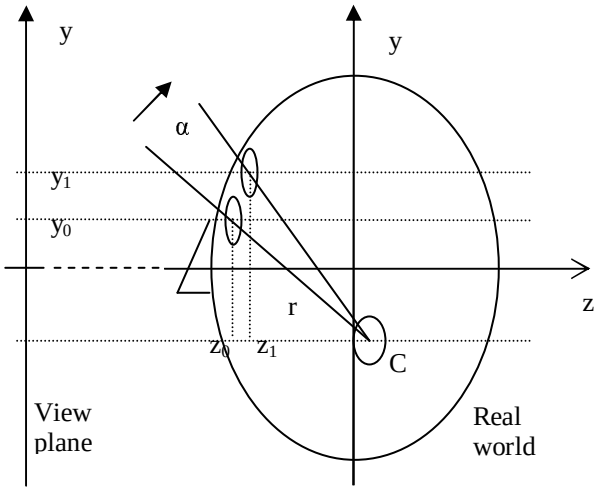


Figure 6. Depth is calculated based on a change in pitch angle.

5.1 Formulas

Let C = (C_x, C_y, C_z) be the centre of rotation of the human head. The values are estimated as (0 ES, -1290 ES, 270 ES). All parameter uses ES (Eye separation) for units, which is defined as the eye distance (in pixel) divides by

1024. The formulas assumed that the origin of coordinate system is at the centre point of the head, and so are the parameters above relative to.

From the change of the y position of the feature point after a head rotation on the pitch angle, we could find the depth of the feature point, z_0. This is depicted in Figure 6, and the derived formulas from the diagram are:

$$(y_1 - c_y) = (y_0 - c_y)\cos\alpha + (c_z - z_0)\sin\alpha$$
$$z_0 = c_z - \frac{(y_1 - c_y) - (y_0 - c_y)\cos\alpha}{\sin\alpha}$$

From the change of the x position of the feature point after a head rotation on the yaw angle, we could find the depth of the feature point, z_0. This is depicted in Figure 7, and the derived formulas from the diagram are:

$$(x_1 - c_x) = (x_0 - c_x)\cos\beta + (c_z - z_0)\sin\beta$$
$$z_0 = c_z - \frac{(x_1 - c_x) - (x_0 - c_x)\cos\beta}{\sin\beta}$$

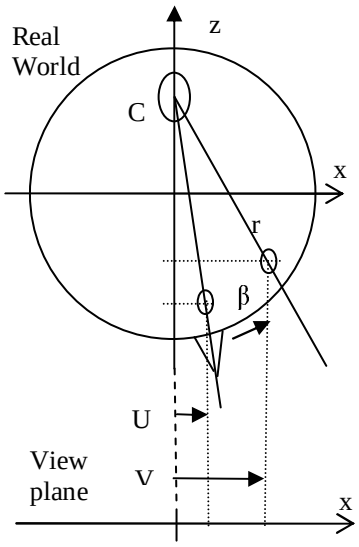


Figure 7. Depth is calculated based on a change in yaw angle.

The depths found from the pitch angle and yaw angle could be different. The depth is combined such that angle with larger magnitude has more influence.

$$z = z_\alpha \frac{\sin\alpha}{\sin\alpha + \sin\beta} + z_\beta \frac{\sin\beta}{\sin\alpha + \sin\beta}$$

where z_α and z_β are the z_0 calculated from α and β respectively.

6 Combine the depth calculation together

Depth calculated by each of the additional images could be different. The depth is combined based on the similarity of the feature matching, smaller the similarity value, heavier the weight is. If the similarity is more than the discard threshold, the feature point is considered as no matching, and it does not participate into the depth calculation. In our example, the discard value is 50.

For a particular feature point, the calculation of the final depth value is:

$$Z = \sum_{i \in I} Z_i \cdot \frac{\sum_{j \in I, j \neq i} S_j}{\sum_{j \in I} S_j}$$

where the set I is the images that the particular feature point is not discarded. Z_i is the depth calculated from the particular image, and S_i is the similarity of the feature matching.

7 Results

The results of the depth map generated are shown in Figure 8, with different dimension of the triangular mesh for comparison. The dimensions are 33 x 37, 49 x 55 and 98 x 110. Black colour denotes points at the background, and lighter the greyscale, closer the feature point is to the camera. The depth of the non-feature points are calculated by triangular interpolation.

The results are not as accurate as we wished. We could see the position of the eyebrows and nose from the result. The lips could also vaguely been seen, but there is a miscalculation near the middle of the lips. There seems to be problems on the border of the head in general. The border of the head should be further away from the camera.

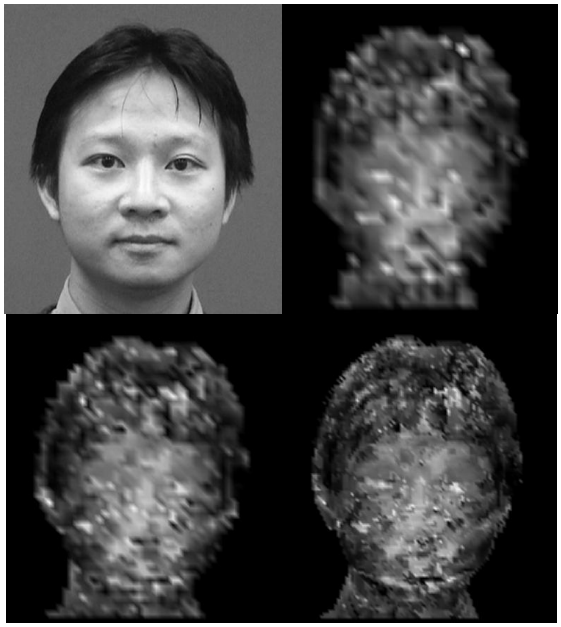


Figure 8. Comparison of the depth maps with different grid dimension: 33 x 37 (top left), 49 x 55 (bottom left) and 98 x 110 (bottom right).

It is possible to use the triangular mesh result to render a face with different head pose. The texture is mapped using the front view image. A few results are shown in Figure 9. The problem of at the border could easily be seen when rendered with a yaw angle. The miscalculations near the lips area could also be noticed when rendered with a pitch angle. The increase of grid dimension increases the quality of the rendered image but it does not solve the mentioned problems. The issue on the head border may be

improved by adding a geometric constraint of the human head. The miscalculation of the lips is caused by inaccurate image registration, which may be improved if a better feature mapping technique is used.

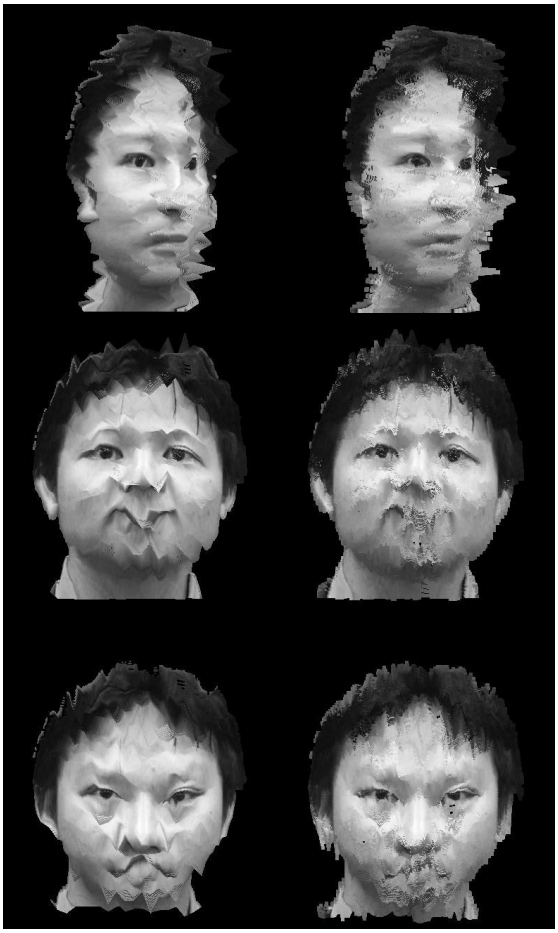


Figure 9. Face rendered with yaw angle of 45˚ (top row), pitch angle of 30˚ (middle row) and pitch angle of -30˚ (bottom row) in different grid dimension: 33 x 37 (left column) and 98 x 110 (right column).

8 Conclusion

This paper shows a way of 3D reconstruction from four images. After head pose determination and image registrations, we could calculate the depth values of the feature points, and hence construct a depth map of the front view of the human head with triangular interpolation. There seems to be two main issues with the result, the problem of estimation near the border of the head, and there is a miscalculation in the middle of the lips. The result is not as accurate as we hope, but it demonstrated that it is possible to have 3D reconstruction of a human face with only four images and without any predefined face model.

9 References

Brand, M; Bhotika, R (2001): Flexible flow for 3D nonrigid tracking and shape recovery, Mitsubishi electric research laboratory, technical report, TR-2001-38. http://www.merl.com. Accessed June 2003.

Brown, L.G. (1992): A survey of image registration techniques. *ACM Computing Surveys (CSUR)*, **24** (4), December 1992.

Gee, A. H.; Cipolla, R. (1994): Determining the gaze of faces in images. *University of Cambridge, CUED/F-INFENG/TR 174.*

Gemmell, J., Zitnick, C.L., Kang, T., Toyama, K., and Seitz, S. (2000): Gaze-awareness for videoconferencing: a software approach. *IEEE Multimedia*, **7**(4):26–35.

Georghiades, A.S. (2003): Measurement and color matching: Recovering 3-D shape and reflectance from a small number of photographs. *Proc. the 13th Eurographics workshop on Rendering,* June 2003:230-240.

Ho, S-Y., and Huang, H-L. (1998): An analytic solution for the pose determination of human faces from a monocular image. *Pattern Recognition Letters*, **19**(11):1045-1054.

Ji, Q; Yang, X (2002): Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-Time Imaging*, **8**(5): 357-377, October 2002.

Lyness, C., Marte, O-C, Wong, B, Marais, P (2001): Image-based techniques in computer graphics: Low-cost model reconstruction from image sequences. *Proc. the 1st international conference on Computer graphics, virtual reality and visualization*, November 2001.

Krishnan, S., Raviv, D. (1995): 2D feature tracking algorithm for motion analysis. *Pattern Recognition*, **28** (8)1103-1126, 1995.

Yip, B, Jin, J.S. (2003): Face re-orientation using ellipsoid model in video conference; *Proc. 7th IASTED International Conference on Internet and Multimedia Systems and Applications 2003,* Aug 2003:245-250.

Yip, B, Jin, J.S. (2004): Viewpoint determination and pose determination of human head in video conferencing based on head movement. *Proc. the 10th International Multi-Media Modelling Conference*, Jan 2004: 130-135.

Ziegler, R, Matusik, W, Pfister, H, McMillan, L (2003): 3D reconstruction using labeled image regions. *Proc. Eurographics/ACM SIGGRAPH symposium on Geometry processing*, June 2003:248-259.

Zitnick, C.L., Gemmell, J., and Toyama, J. (1999): Manipulation of video eye gaze and head orientation for video teleconferencing. Microsoft Research, Technical Report, MSR-TR-99-46.