Journal of Theoretical and Applied Information Technology
30th November 2012. Vol. 45 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

FACIAL FEATURE LOCATING USING ACTIVE APPEARANCE MODELS WITH CONTOUR CONSTRAINTS FROM CONSUMER DEPTH CAMERAS

[1, 2]QINGXIANG WANG, [2]XIAOQIANG REN

[1] School of Computer Science and Technology, Shandong University, Jinan, 250100 Shandong, China

[2] School of Information, Shandong Polytechnic University, Jinan, 250353 Shandong, China.

ABSTRACT

Active appearance model (AAM) is a powerful method for objects feature localization and successfully used in computer vision. Although widely employed, AAMs suffer from a few drawbacks, such as accuracy on the contour of face. We present an Active Appearance Models with contour constraints method from depth data acquired by consumer depth cameras, which could capture video data in 3D, include depth data and color image, under any ambient light conditions based on invisible infrared projection. The facial contour points are acquired from depth data that can be used to restrain the fitting result of AAM and improve the accuracy of the contour part of the result shape by minimize the Euclidean distance in parameter space from the previous state to the solution. Experimental results show the improvement on the facial feature locating.

Keywords: *Facial Feature, Active Appearance Models, Consumer Depth Cameras*

1. INTRODUCTION

Active appearance model (AAM), proposed by Cootes et al[1], is a powerful method for objects feature localization and widely used in face feature localization, identify[2] video tracking[3], attitude estimate[4, 5] of medical image processing, etc. AAM is a model based pattern recognition method, comes from Snake[6] and ASM (active shape model) [7]. Unlike only concerned the local features in ASM, AAM use the shape and texture information to model the deformable objects and not easily into the local minimum value.

Because the traditional AAM method use the texture residual of shape delaunay triangulation as the basis for the optimization, it will incline to fit the internal points better in the fitting process. The contour features points are sparser and the texture information for training and fitting only on one side of the vertex, so usually the contour features of shape result are not very well to joint the actual contour, especially when the training samples are insufficient or illumination conditions are varied. However the ASM which use the local information is better than the AAM on contour points, some progress combine ASM and AAM to improve accuracy[8] but still have the ASM restrictions on the local minimum. And others use the 3D reconstruction or 2D+3D method[9, 10] to improve the accuracy, but they need a continuous data or stereo camera to construct the 3D model.

In last several years, 3D capture devices appeared and could acquire depth data with the color data of the real scene. With the depth data the contour points could be easy acquired from the background and tracked. In this paper we will use one of those devices, Kinect, to capture depth data and restrain the fitting result of AAM.

We present a method for facial feature locating under the depth data captured from consumer depth cameras and improve the contour part of the AAM accuracy by contour constraints.

This paper is organized as follows. In section 2, the Active Appearance Models algorithm is introduced, and the definition of facial points is shown. In section 3, we give the data acquisition from consumer depth cameras, and the initialization, and present the fitting process with contour restraint. The experimental results proposed in this paper are also presented in this section. Finally, our work of this paper is summarized in the last section.

2. ACTIVE APPEARANCE MODELS

The Active Appearance Models (AAMs) [1] is a successful method for matching the feature points of new facial images and is applied in many applications. The method uses the statistical models and the input image's appearance to estimate the shape of the face. In the paper we will improve the AAM algorithm with depth restraint using the consumer depth cameras.

The shape of an AAM is defined by a 2D triangulated mesh and in particular the vertex locations of the mesh. Mathematically, the shape S of an AAM is defined as the concatenation of the x and y-coordinates of the n vertices that make up the mesh: $S = (x_1, y_1, \ldots, x_n, y_n)^T$. A compact model that allows a linear variation in the shape is given by,

$$S = S_0 + \sum_{i=1}^{m} p_i S_i \tag{1}$$

where the coefficients pi are the shape parameters and The base shape S_0 is the mean shape and the vectors S_i are the eigenvectors corresponding to the m largest eigenvalues.

The appearance of the AAM is defined as the pixels $x = (x, y)^T$ that lie inside the base mesh S_0. A compact model that allows a linear variation in the shape is given by,

$$A = A_0(x) + \sum_{i=1}^{l} q_i A_i(x) \tag{2}$$

Where the coefficients q_i are the appearance parameters. A_s with the shape, the base appearance A_0 and appearance images A_i are usually computed by. The base appearance A_0 is the mean shape normalized image and the vectors A_i are the eigenvectors corresponding to the l largest eigenvalues.

After applying PCA to the shape and appearance of the training images, we can get the S_i and A_i and the Relation matrix R between the variation of coefficient and appearance residual. When an image of face is inputted, AAM minimizes Equation (3) to fit a shape to the face by iterative model refinement.

$$\sum_{x\in S_0}\left[A_0(x)+\sum_{i=1}^{m} q_i A_i(x) - I(W(x;p)\right]^2 \tag{3}$$

Every shape in train image is composed of a fix number of points and the final result is the same. In this paper, we use the same definition as XM2VTS11 frontal data, 68 points (see figure 1). The points 0-14 are the points on the contour of face.

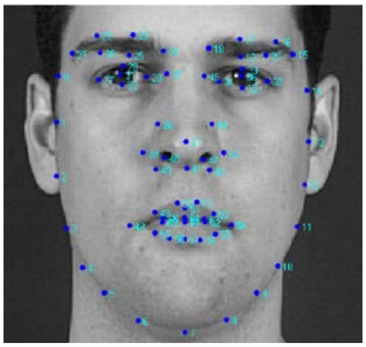

Figure 1: Facial Feature Points (68 Points)

3. FITTING PROCESS WITH CONTOUR RESTRAINT

In this section, we give the data acquisition from consumer depth cameras, the initialization method, and present the fitting process with contour restraint. The experimental results proposed at last.

2.1 Data Acquisition

The input data is acquired from the Kinect, the hardware which could capture which captures video data in 3D under any ambient light conditions based on invisible infrared projection. That is a low-cost acquisition device and has some essential benefits such as ease of deployment and sustained operability in a natural environment.

We employ the Kinect for getting the data which gives a 640x480 image with depth resolution of a few centimeters with the SDK published by Microsoft (figure 2).

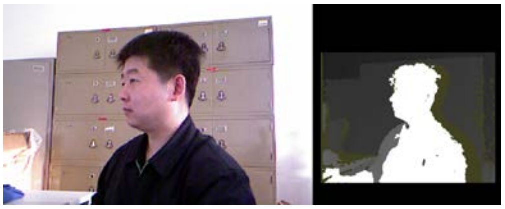

Figure 2: Captured Data From Kinect

With the captured data, we align the 2D RGB image with the depth data and save to files.

In this paper the user is about 1.2m in front of the Kinect sensor, face to the camera, and the face region in the image is about 100x100 pixels.

2.2 Initialization

When a image with depth data from Kinect is coming, we first use the Viola-Jones face detect method[12] to find the region of the face, this region we defined as Sin and then initialize the initial value of AAM iterative coefficients with it. Initialization with a given region is to adjust the size and position of S (see (1)) to make bounding rectangle of S to fit the region.

Because of the depth data may provide more accurate initialization, after initializing the initial value with face detect, we use the threshold to get the head region on the depth map. In this paper we use the mean value of the depth of the face detect area and the threshold is set to 30, see the definition below,

$$HeadRegion = \left\{ \begin{array}{l} x; \left| d(x) - D \right| \le 30, \\ D = \frac{1}{n} \sum\limits_{y \in S_{in}} d(y), \\ d(y) \in \left[800,1500 \right] \end{array} \right\} \qquad (4)$$

where d(x) is the depth value of the pixel x, n is the number of pixels which is in initial area Sin. The face detect region will include some pixels on the background or something that is not contained in the head; we choose the 800 to 1500 as the limits.

With the head region, we calculate the minimum bounding rectangle (MBR), and adjust the MBR size to fit the face area. After experiment, we decide to cut 1/7 from the top of the head and 1/10 from the bottom and then use the left region to initialize the initial value of AAM, see figure 3b.

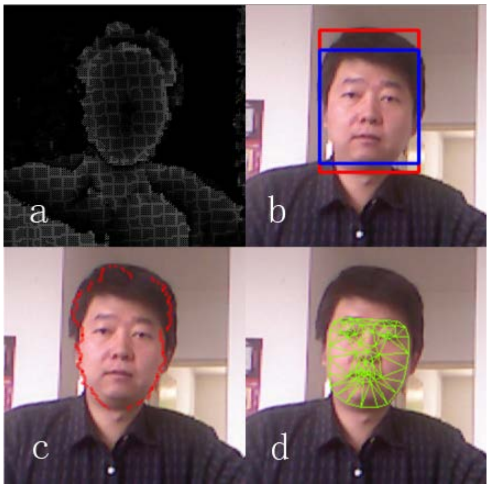

Figure 3: (A) Depth Image, (B) RGB Image And Red Rect Is MBR And Blue Rect Is Face Area, (C) Contour Points Acquire D From Depth Data, (D) Fit Result

2.3 Fitting Process

The idea of the fitting process is as follows. Firstly, with initial value acquired in previous section, the algorithm iterates using the original AAM to find the coefficients of shape which are close to the final result. Secondly we get the contour of face with the face region and then find the nearest points of every feature points on contour. In the following constrain the shape to the contour at next iteration and finally the facial feature with contour restraint will be given.

In the first step, we follow the method of T.F.Coots to minimize Equation (3) with the initial p_i and q_i from section II (with a given image and initial shape we could get the coefficients of AAM by reverse process). In this step, the result is composed of shape parameter p_f, appearance parameter q_f and global transformation and the first two are both linear conversion. For the convenience of expression, we combine them as one parameter $c_f = \{ p_f, q_f \}^T$.

The second, we use the depth data to get the contour of the head region. In section II, the area of head has been acquired and then we start from the centre and draw radial to the points on MBR on depth map and save the points whose gradients are biggest on the radials (figure 3c). Then on the contour we find the points nearest to the markup 0-14 of the shape result in first step.

At last, our approach acquire the final result of the facial features by following that of Lewis and Anjyo[13] and J. Rafael Tena[14] in which the user is allowed to click and drag points to manipulate vertices on the mesh of the face model and constrain them to a desired location. We take these points V_k as the click and drag points. But when only use these points, we find that the internal point deviated from its correct position because of stretching. So unlike their method, this approach added canthus and mouth corners as constraint feature points (points 27, 29, 32, 34, 48, 54 in figure 1) in addition to these contour points and record the x and y-coordinates $(x_0, y_0, \ldots, x_{14}, y_{14}, x_{27}, y_{27}, \ldots, x_{54}, y_{54})^T$ as V_k.

$$E(c) = \sum_{k=1}^{K} \left\| V_k - S_k c \right\|_2^2 + \gamma \left\| c_f - c \right\|_2^2 \qquad (5)$$

Where v_k is the kth point of V_k, K is the number of constraint feature points. S_k are AAM bases, see in Equation (1)， c_f is the coefficients of AAM by previous iteration step, c is the shape coefficients which minimize the Euclidean distance in parameter space from the previous state to the solution, γ is the weight. A lower value of γ maintain the origin AAM fit result. Conversely, a high makes the model adapt the contour points and may distort the inner points of the shape. In this paper we set it to 10. Because of V_k are 15 coutours points and some selected feature points, only part of the S_k need to be multiplied in Equation (5).

We minimize E(c) by finding the point at which the gradient vanishes. The partial derivative of Equation (5) yields

$$\sum_{k=1}^{K}\left(s_k^{T} s_k c - s_k^{T} v_k^{T}\right)+\gamma(c-c_f)=0 \quad (6)$$

This function can be directly solved by Gauss elimination.

2.4 Experimental Results

For feature point training, we use the Manchester and IMM Face Database and add a few pictures to adapt to the Asian by the tools am_tools_win_v3 downloaded from Tim Cootes's web site of Manchester. All the experiments are implemented with C++ and OpenCV2.1 on PC with Intel(R) CPU, 2.33GHz, 2G Memory. We tested our method on 20 images from 2 videos captured from Kinect with that the subject is about 1.2 m in front of the Kinect sensor. Some results are showed in fig 4.

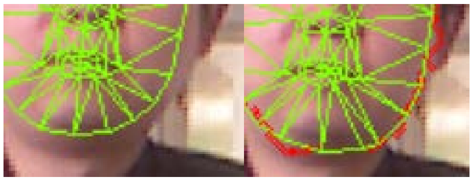

Figure 4: Result. Left Is Origin AAM And Right Is Ours. The Features Of Contours In Right Are Closer To The Facial Contour, Red Dot Are The Points Of Contours And The Vertex Of Green Trimesh Are The Facial Shape Result.

In this figure, the red dot are the are the points of contours which are acquired with segmentation by depth data(see 3.2),

In table I, we show the accuracy of the facial feature locating result in the frames inputted. The RMSE is calculated by below,

$$RMSE(x,x_g)=\frac{1}{n}\sum_{n=1}^{n}\left(\left(x_i-x_{g,i}\right)^2+\left(y_i-y_{g,i}\right)^2\right)^{1/2} \quad (7)$$

In this equation, n is the number of the facial feature points, x is the fit result and x_g is ground truth which is labeled manually in every frame. We calculate the RMSE of inner points and coutour points separately.

The inner points are the points from 15 to 67, and the contour points are 0-14. The points

Table I: Residual Errors For Generic Ground Truth Experiments.

	RMSE (residual error per vertex, pixels)			
	original		ours	
Position	Inner	Contour	Inner	Contour
Error	*1.6*	*2.2*	*1.7*	*0.9*

By testing, the result could be seen in figure 4 and table I, that our algorithm are better fit at contour but a little weak on the inner features.

4. CONCLUSION

We present a method for facial feature locating under the depth data captured from consumer depth cameras and improve the AAM accuracy by contour constraints. The contour points are acquired from depth data synchronously with the RGB image which contains a front view human face. With that we set a threshold to get the contour of facial boundary and improve the accuracy of AAM on the contour features. The experiment results show that our algorithm has a reasonable fit accuracy on part of the facial markups. However the accuracy of inner features are a little decrescent that would be caused by the global constrain process such that the inner features would be dragged from the original position to satisfy the contour features. Segment the facial region or nonlinear stretch with distance would be useful. These could be the future work of the method.

ACKNOWLEDGEMENTS

This work was supported in part by the Natural Science Foundation of Shandong Province (NO.ZR2011FQ033), Ji'nan Youth Science and Technology Star Project (No.20110302), A Project of Shandong Province Higher Educational Science and Technology Program (NO.J09LF23, NO.J09LG07), and Institute Support Innovation Project of Jinan Administration of Science &Technology (No. 201202012)

REFRENCES:

[1] T. F. Cootes,G. J.Edwards, and C. J.Taylor, “Active appearance models”, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 6, 2001, pp. 681–685.

[2] Liang, L., Xiao, R., Wen, F., and Sun, J, “Face alignment via component-based discriminative search”, Proceedings of the 10th European Conference on Computer Vision (ECCV 2008), Springer, October 12-18, 2008, pp. 72-85.

[3] Matthews, I., Xiao, J., and Baker, S, “2D vs. 3D deformable face models: Representational power, construction, and real-time fitting”. International Journal of Computer Vision, Vol.75, No.1, 2007, pp. 93–113.

[4] Murphy-Chutorian, E. and Trivedi, M. M., “Head pose estimation in computer vision: A survey”. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.31, No.4, 2009, pp.607–626.

[5] R. Beichel, H. Bischof, F. Leberl, and M. Sonka, “Robust active appearance models and their application to medical image analysis”, IEEE Transaction on Medical Imaging., Vol. 24, No. 9, 2005, pp. 1151–1169.

[6] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models”, International journal of computer vision, Vol.1, No.4, 1988, pp. 321–331

[7] T. F. Cootes, D. H. Cooper, C. J. Taylor, and J. Graham, “Active shape models -Their training and application”, Computer Vision and Image Understanding, Vol. 61, No. 1, 1995, pp. 38–59.

[8] J. Sung, T. Kanade, and D. Kim, “A unified gradient-based approach for combining ASM into AAM”, International Journal of Computer Vision, Vol. 75, No. 2, 2007, pp. 297–309.

[9] J. Xiao, S. Baker, I. Matthews, and T. Kanade, “Real-time combined 2D+3D active appearance models”, Proceedings of IEEE Society Conference on Computer Vision and Pattern Recognition, IEEE Conference Publications, June 27, 2004, pp. 535-542.

[10] J. Liebelt, X. Jing, and Y. Jie, “Robust aam fitting by fusion of images and disparity data”, Proceedings of IEEE Society Conference on Computer Vision and Pattern Recognition, IEEE Conference Publications, June 17-22, 2006, pp. 2483-2490.

[11] K. Messer, J. Matas, J. Kittler, J. Luettin and G. Maitre. “XM2VTSbd: The Extended M2VTS Database”, Proceedings 2nd Conference on Audio and Video-base Biometric Personal Verification, Springer, March, 1999, pp.72-77.

[12] P. Viola and M. J. Jones. “Robust real-time face detection”. International Journal of Computer Vision, Vol. 57, No. 2, 2004, pp.137–154.

[13] Lewis, J. P., and Anjyo, K., “Direct manipulation blendshapes”, Computer Graphics and Applications, Vol. 30, No. 4, 2010, pp.42 – 50.

[14] J. Rafael Tena, Fernando De la Torrey, Iain Matthews. “Interactive Region-Based Linear 3D Face Models”, ACM Transactions on Graphics-Proceedings of ACM SIGGRAPH 2011, Vol 30, No. 4, 2011, pp.1-10.