FACIAL POSE ESTIMATION FOR IMAGE RETRIEVAL

Andreas Savakis and James Schimmel
Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA

Keywords: Pose estimation, image retrieval, neural networks, eye detector, mouth detector.

Abstract: Face detection is a prominent semantic feature which, along with low-level features, is often used for content-based image retrieval. In this paper we present a human facial pose estimation method that can be used to generate additional metadata for more effective image retrieval when a face is already detected. Our computationally efficient pose estimation approach is based on a simplified geometric head model and combines artificial neural network (ANN) detectors with template matching. Testing at various poses demonstrated that the proposed method achieves pose estimation within 4.28 degrees on average, when the facial features are accurately detected.

1 INTRODUCTION

Semantic information in the form of face detection is often included in content-based image retrieval systems. When a face is detected in the query image, additional information, such as facial pose estimation, would be very helpful for more effective image retrieval. This paper presents a computationally efficient pose estimation method that can be used to generate additional metadata given that a human face is already detected in the query image.

Faces in images rarely appear frontal and upright, and this fact underscores the importance of reliable, computationally efficient pose estimation algorithms that aid image retrieval methods. While a face can undergo pitch, roll and/or yaw rotations, this paper focuses on yaw angle estimation which is most commonly encountered. The ‘yaw’ of a face is the angle at which the face turns horizontally right or left and has the greatest possible benefit to applications, since it is an indication of which way a person is looking or heading.

Pose can be estimated using various methods, however, the most computationally efficient approaches tend to be geometric. In this paper, we propose a novel geometric head model approach, where proportions between the facial features are used to calculate the pose angle. Section 2 presents the background and assumptions used for the proposed method. Section 3 discusses facial feature detection using ANN’s and is followed by the pose estimation method based on a simplified geometric head model in Section 4. Sections 5 and 6 present results and conclusions respectively.

2 BACKGROUND

An important aspect of geometric pose estimation methods involves the human facial features to be used. Selecting the two eyes as feature points is useful, because eyes are prominent and relatively easy to detect. The mouth is another prominent feature, but its shape can vary depending on the person’s activity. The nose can be harder to detect due to its lack of contrast, however, some methods (Burl and Perona 1996), (Choi et al. 1998) make use of this feature with proper filtering. Facial feature detection is often the most computationally demanding portion of a pose estimation algorithm (Fergus et al. 2003). To reduce computations, we limit the number of features to three (left eye, right eye and mouth), which reduces complexity while maintaining accuracy.

Our method employs ANN’s for feature detection and exploits the angles of a geometric triangular template formed by the detected eyes and mouth features. Other methods that have used ANN’s tend to be mainly appearance-based (Rowley et al. 1998), (Haddadnia et al. 2002). These methods train their networks on small images of the whole face turned at different angles, while the proposed method searches for individual facial features.

Savakis A. and Schimmel J. (2007).
FACIAL POSE ESTIMATION FOR IMAGE RETRIEVAL.
In *Proceedings of the Second International Conference on Computer Vision Theory and Applications - IU/MTSV*, pages 173-177

Triangle or pyramidal templates were presented in (Choi et al. 1998) and (Yilmaz and Shah 2002), where these methods perform pose estimation using a classification system based on template deformation or orthogonal projections of the detected points. Our method focuses on the efficiency of detecting pose by using a small number of feature points (three) and by directly determining pose from two of the angles of the triangular template.

The following simplifying assumptions were made for easily training feature detectors: (a) facial expression is assumed to be calm with the eyes open; (b) there are no facial occlusions, such as hats, glasses, facial hair, etc. Violation of these assumptions would make the detection of eyes and mouth more difficult, but once these facial features are detected, the geometric pose estimation would still work. The final assumption is that both eyes are visible, which limits the pose angle to ±30 degrees. The next sections cover the detection of the facial features used, present the head model and discuss the results.

3 FACIAL FEATURE DETECTION

The first step in the detection of facial features is done by detecting the face using a face detection method and narrowing the search space of the facial feature detectors based on the skin region. Once a face is detected, the skin detection method selected to narrow the search space is a lookup table based on pixel probability distributions in the YCbCr color space. The most notable challenge with skin detection is due to lighting conditions (Storring et al. 1999), (Gong et al. 2000). The luminance portion of the image is ignored here, which makes the process more resilient to lighting effects.

With pose estimation, facial features undergo deformations as the pose angle changes. This and the variability in facial proportions from person to person, make the feature detection problem challenging. Previous approaches have involved the segmentation of the facial image and then the use of shape detectors to determine the location of the features (Storring et al. 1999). Areas of high change are often related to facial features (Fitzpatrick 2003), which motivates using edges in a facial image or looking at changes in pixel intensities.

Artificial Neural Networks (ANNs) are used here for the detection of facial features from grayscale images. An eye detection network was trained using 2760 eye images and 13800 non-eye images of size 21x11. The training images for the eye network consisted of both left and right eye images at varying poses up to 30 degrees. A mouth detection network was trained using 1430 mouth images and 7150 non-mouth images of size 33x13. The eye network provides accuracy of 90%, while the mouth network results in accuracy of 93%.

During feature detection, the image is scaled based on the results from the skin detection process. The grayscale image is then subjected to contrast stretching for normalization. Each detector network is applied through the skin area of the image using a sliding window and produces an output result between 0 and 1 for each pixel. The higher the output value, the more likely the subimage centered at the pixel of interest contains a feature. Two weight maps are generated for the location of eyes and the mouth respectively. An averaging filter is passed over the maps to reduce the effects of outliers. Local maxima indicate likely locations of the eyes and mouth.

Each combination of likely feature points is checked to make sure they adhere to template limitations. Finally, the set with the greatest total weight is selected as the correct position for the facial template.

4 POSE ESTIMATION

A simplified head model considered for fast pose estimation is shown in Figure 1. Note that the eyes and mouth are used as the primary feature points. The head itself is treated as a spherical object of radius *r* which rotates upon the y-axis. The mouth and eyes are treated as a vertical plane on this sphere. The distance between the eyes is labeled *d.*

By projecting the model of the head onto a two dimensional plane, Equations (1), (2) and (3) were developed to describe the change in the triangular template angles, α_1'', α_2'', and α_3'', as the model increases in yaw pose angle.

$$\alpha_1'' = \arctan\left(\tan(\alpha_1) * \frac{\sin\theta_0}{\sin(\theta_T) - \sin(\theta_T - \theta_0)} \right) \tag{1}$$

$$\alpha_2'' = \arctan\left(\tan(\alpha_2) * \frac{\sin\theta_0}{\sin(\theta_0 - \theta_T) - \sin(\theta_T)} \right) \tag{2}$$

$$\alpha_3'' = \pi - (\alpha_1'' + \alpha_2'') \tag{3}$$

where the pose angle is represented by θ_T, variables α_1, α_2, and α_3 represent the angles of the triangular face template in the frontal position, and variable θ_0 represents the angle formed by the eyes and the central axis of rotation (Figure 1b). It is important to note that since the sum of the angles in the triangular template is 180 degrees, only two of the angles are needed to fully describe it within a similarity transformation. Figure 2 shows a graph of these theoretical angle changes with respect to pose angle.

The quantities $\theta_0, \alpha_1, \alpha_2$, and α_3 all vary from individual to individual making it necessary to normalize the angle relative to one another when performing the final pose estimation. Figure 3 is a graph of $90-\alpha_1$ versus $90-\alpha_2$, the two top angles of the triangle, as the face pose changes. The figure shows three theoretical curves and two curves generated from image sets. The lines from the origin connect points on the curves at -30°, 0° and 30°. This graph helps illustrate how the pose angles form curves around a given focal point. This is due to the symmetry inherent in the human face. At the forward position, these two angles are equal and we are operating on the graph diagonal. When the pose changes, the difference between these angles increases, as illustrated in the curves shown.

The pose angle is equal to the angle formed by the center line, where the face is in the frontal position, and the line formed by the calculated α_1, α_2 point and the origin. Using this observation and the previous equations, a simplified equation for the pose can be developed:

$$\theta_T = \arctan\left(\frac{\alpha_2 - \alpha_1}{\alpha_3}\right) \tag{4}$$

This equation is easy to compute once the facial features have been detected and the template triangle has been formed.

5 RESULTS

In order to test the pose estimation algorithm an image database was generated under controlled conditions. Images of ten individuals were obtained at various poses taken at 5° increments from 30° to -30°, with 0° being considered the frontal position. Samples of the facial images from this database are shown in Figure 4. None of these images were used in the training of the ANN's. Lighting, facial expression, and head movement were kept constant.

At first, the center locations for the right eye, left eye and mouth were selected manually. From these values the pose was calculated independently on each of the test images. With most image sets the error tends to be lowest at the frontal position and worsens gradually as the head approaches the extreme angles. Table 1 summarizes the results by showing the error from all ten test sets. The average error for all datasets was 4.28 degrees.

Results based on automatic feature detection were obtained using all 130 images. The algorithm was able to detect the features and apply a template correctly for 120 of the images or a 92.3% correct detection rate. Problems with the automatic scaling or a person's particular features caused most of the missed detection.

The automatic detection method performed well compared to the best possible performance obtained by manual methods, as shown in Table 1. From this table the average error due to automatic feature detection was 6.41° degrees.

The entire process can be optimized using Intel's Synchronous SIMD Extensions (SSE) (Farber 2003). After optimization, pose estimation only requires 141ms, when processing an image on a 1.8 GHz Pentium 4 workstation, which renders the method suitable for real-time applications.

6 CONCLUSIONS

The pose estimation method presented in this paper offers both robust and efficient pose estimation that can be incorporated in content-based image retrieval systems. The accuracy of the algorithm makes this method capable of automatically determining the pose of a human face to within 10° of the actual value, while limiting the amount of computations that must be performed on the image.

Currently this method is designed to determine pose in only the yaw direction. Future work should expand the model to include ways of determining changes in the roll and the pitch of the head and to incorporate pose estimation in content-based image retrieval systems.

ACKNOWLEDGEMENTS

This research was sponsored in part by the Eastman Kodak Company and the Center for Electronic Imaging Systems (CEIS), a NYSTAR-designated

Center for Advanced Technology in New York State.

REFERENCES

Burl M.C., and Perona P., 1996. Recognition of Planar Object Classes. In *IEEE Conference on Computer Vision and Patten Recognition*.

Choi K. N., Carcassoni M., and Hancock. E., 1998. Estimating 3D Facial Pose using the EM Algorithm. In *9th British Machine Vision Conference*.

Fergus R, Perona P., and Zisserman A., 2003. Object Class Recognition by Unsupervised Scale-Invariant Learning. In *IEEE Conference on Computer Vision and Pattern Recognition*.

Rowley H. A., Baluja S., and Kanade T., 1998. Neural Network-Based Face Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 23–38.

Haddadnia J., Faez K., and Ahmadi M., 2002. N-Feature Neural Network Human Face Recognition. In *15th international Conference on Vision Interface*.

Yilmaz A. and Shah M., 2002. Automatic Feature Detection and Pose Recovery for Faces. In *5th Asian Conference on Computer Vision*.

Storring M., Andersen H. J. and Granum E., 1999. Skin Colour detection under changing lighting conditions. In *7th symposium on Intelligent Robotics Systems*.

Gong S., Mckenna S. J., and Psarrou A,, 2000. *Dynamic Vision*. Imperial Collage Press, London.

Fitzpatrick P., 2003. Head Pose estimation without manual initialization. *Massachusetts Institute of Technology. AI Lab Technical Report*.

Farber A., 2003. Introduction to SSE Programming. http://www.codeproject.com/cpp/sseintro.asp. The Code Project.

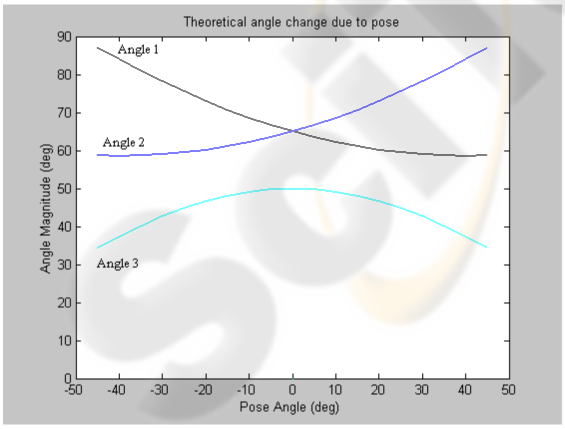
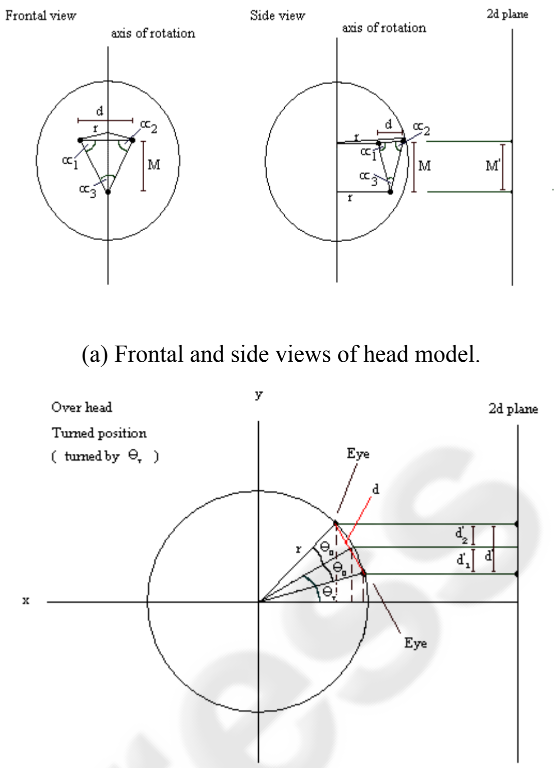


Figure 2: Graph of theoretical equations of template angles vs. pose variation.



(a) Frontal and side views of head model.

(b) Overhead view of head model.

Figure 1: Head model using eyes and mouth as feature points that generate a triangular template.

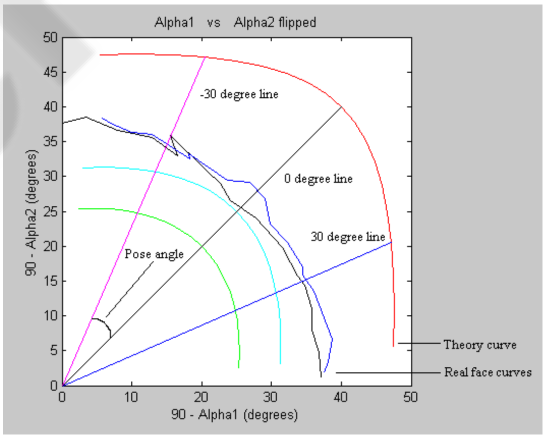


Figure 3: Graph used for pose estimation, where pose angle is the angle of deviation from the diagonal line.

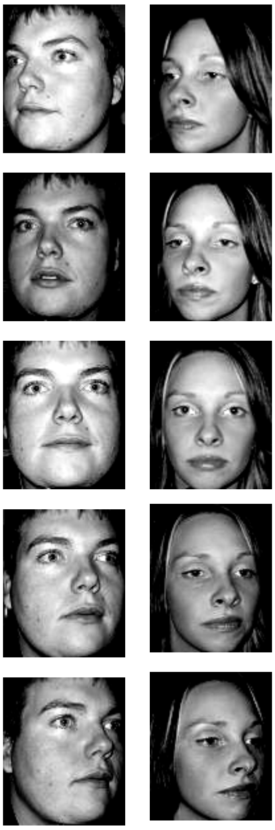


Figure 4: Sample images used for pose estimation.

Table 1: Pose estimation results for manual feature detection and automatic feature detection.

Pose Estimation Error Pose between -30 and 30 degrees		
	Manual feature detection	Automatic feature detection
Image Set	Average Error (degrees)	Average Error (degrees)
1	4.42	4.43
2	2.62	4.01
3	2.23	4.33
4	6.04	11.80
5	7.48	9.19
6	4.48	7.65
7	4.05	3.83
8	3.94	4.86
9	5.15	8.57
10	2.39	5.44
Overall	**4.28**	**6.41**