Face pose estimation from eyes and mouth

This item was submitted to Loughborough University's Institutional Repository by the/an author.


Additional Information:

- This article was published in the journal, International Journal of Advanced Mechatronic Systems (IJAMECHS) [© Inderscience Enterprises Ltd] and the definitive version is available at http://dx.doi.org/10.1504/IJAMECHS.2010.030857

Metadata Record: [https://dspace.lboro.ac.uk/2134/10174](https://dspace.lboro.ac.uk/2134/10174)

Version: Accepted for publication

Publisher: © Inderscience Enterprises Ltd.

Please cite the published version.
This item was submitted to Loughborough’s Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
Face Pose Estimation from Eyes and Mouth

M. Shafi*
Department of Computer Science,
Loughborough University, UK
E-mail: m.shafi@lboro.ac.uk
*Corresponding author

P.W.H. Chung
Department of Computer Science,
Loughborough University, UK
E-mail: p.w.h.chung@lboro.ac.uk

Abstract: Face pose estimation plays an important role in human computer interaction, automatic human behavior analysis, gaze estimation, virtual reality, pose independent face recognition etc. Accuracy and speed are the most desirable features of a face pose estimation system. In this paper, a face pose estimation scheme based on the centers of the eyes and mouth is proposed. The proposed method is simple and is, therefore, very effective in terms of computation because it uses only three points i.e. eyes and mouth centers. The use of only three points increases the pose estimation range and makes the method suitable for real time applications. Tests using the Pointing’ 04 database show that the proposed scheme is robust and fast.

Keywords: human-computer interaction; mouth map; eye map; pose-independent pose estimation.


Biographical notes: Muhammad Shafi completed his Bachelor in Computer System Engineering from Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan in 2005. He is currently PhD candidate in the Department of Computer Science Loughborough University, UK.

Paul W.H. Chung is a Professor of Computer Science at Loughborough University, UK. He received his B.Sc. in Computing from Imperial College London in 1981 and Ph.D. in Artificial Intelligence from the University of Edinburgh in 1986. From 1984 to 1991, he worked at the AI Applications Institute, University of Edinburgh and joined Loughborough University in 1991.

1 INTRODUCTION

Facial pose estimation plays an important role in pose-independent face recognition, gaze estimation, virtual reality applications and human computer interaction etc. As suggested by Murphy-Chutorian and Trivedi (2008), an ideal pose estimation system should:

- **Accuracy**: provide a reasonable estimate of the face pose with minimum error.

- **Monocular**: be able to work with images taken with a monocular camera because they are more suitable for real life applications.

- **Autonomous**: be completely autonomous. There should not be any manual initialization, feature selection, etc.

- **Identity Invariant**: The system should work across all identities.

- **Real-time**: be able to estimate a continues range of head orientation in real time.
• **Range:** be able to work for a whole range of poses, even if the face is directed away from the camera.

• **Resolution-Independent:** The system should apply to both low and high resolution images.

In the proposed method, some of the above features have been considered. The proposed method has the following advantages over the other pose estimation systems in literature:

• It is based on mouth and eyes which are the most stable features of face.

• It is based only on three points which makes it less expensive computationally and hence suitable for real-time applications.

• Since center of mouth and eyes can be calculated without the explicit exposure of eyes and mouth corners, this increases the range of pose estimation.

A typical pose estimation comprises Yaw, Pitch and Roll estimation of a face. The main target of the proposed method is Yaw estimation which is the requirement of most real life applications e.g. interactive games, virtual reality applications and driver vigilance assessment, etc. However, to be suitable for real life applications, the proposed method is capable of working with -15 to 15 degrees of pitch variations. The rest of the paper is organized as follows: section 2 gives an overview of the related work from literature; sections 3 and 4 describe the detection of eyes and mouth centers respectively; section 5 gives the details of pose estimation algorithm; section 6 reports the experimental results; section 7 concludes the paper with future work.

### 2 RELATED WORK

Face pose estimation systems can be divided into two main categories: intrusive and non-intrusive. The intrusive methods require some hardware attached to the person. Although the intrusive methods are more accurate, they are not suitable for many practical applications because of the cumbersome use of hardwares. Non-intrusive methods require no hardware attached to the person and the image or video is taken from a remote camera. Non-intrusive methods can be further classified into feature-based and model-based methods. Feature-based methods are based on the relationships between the face pose and face features such as grayscale, texture etc. Model-based methods use the geometrical structure and prior knowledge of human faces. Facial features are extracted and then compared with a 3D model of a face to estimate the face pose. Murphy-Chutorian and Trivedi (2008) presents a very comprehensive survey of pose estimation methods. An overview of several pose estimation systems is presented in the following paragraphs.

Most of the face pose estimation algorithms have been developed for pose tracking in video sequences. Gordon (1998) proposed a pose estimation method based on the head-motion in videos. Yang et al. (2004) proposed best shot selection using regression algorithm to train a face pose estimator. A head pose estimation with a 3D cylinder model for the face shape has been proposed by Seo et al. (2004). Motwani and Ji (2001) presented a 3D face recovery algorithm using wavelet transform and principal component analysis. Yao et al. (2001) proposed pose estimation using an ellipse-circle technique based on the affine transformation between the features. Seitz and Dyer (1996) proposed a pose estimation method called view morphing which can be applied to both calibrated and uncalibrated images.

Some pose estimation methods work on single images instead of videos. Gee and Cipolla (1994) use five feature points, nose tip and far corners of the eyes and mouth to estimate face pose. The drawback of their method is that generally it is not easy to detect the nose tip accurately in most facial images. Ho and Huang (1998) presented an analytical solution for the pose estimation from monocular image. Wang and Sung (2001) proposed pose determination of human faces by using vanishing points. They have used the corners of the eyes and mouth to derive equation for pose estimation. The proposed method is similar to Ho (1998) and Wang (2001) to some extent. However, it has certain advantages over them. Instead of using eyes and mouth corners, which requires six reference points, the proposed method uses the centres of the eyes and mouth, which requires only three points. Focusing on the centres also increases the range of pose estimation and simplifies the equations used.

### 3 EYES DETECTION

Before searching for the eyes, it is assumed that the face boundary is already given. The method described by Hsu et al. (2002) is used to detect the coarse location of the eyes in a facial image. To do this the input color image is converted to YCbCr plane. An eye map is built from the Cb and Cr components of the YCbCr image, which is based on the observation that high Cb and low Cr values are found around the eyes. If Cb is divided by Cr then the eyes are obtained as white region as shown in Figure 1.

The proposed pose estimation method is very sensitive to the accuracy of the center of the eyes. The method described by Hsu et al. (2002) is not adequate for this purpose. Therefore, two improvements are proposed. The first is an algorithm for adoptive thresholding and the second is a new equation for finding the centers of the eyes. The adoptive thresholding algorithm begins with thresholding the CbOverCr image with a threshold value of zero. The threshold value is incremented until at least two white regions that each approach the size of an eye is obtained. The rectangular region including these white regions give the approximate locations of the eyes. Further thresholding of
Figure 1: Approximate Eye Region Detection

Figure 2: Eyes Corners Detection

this rectangular region is done to extract the corners of the two eyes. Figure 1 shows how the approximate locations of the eyes are obtained. Figure 2 shows how the corners of the eyes are obtained by further thresholding the small region obtained in Figure 1. The adoptive thresholding algorithm in pseudo code is:

input: CbOverCr Image
output: Eyes Corners
Start

ThresholdValue=0
while(1)

Threshold the Image with thresholdValue
ThresholdValue+=increment
if (There are at least two white regions
Stop
Else
Continue

im2=Get the rectangular region around the white regions obtained earlier
Thresh=0
while(1)

Threshold im2 with Thresh
Thresh+=inc
if (white region>15% of im2)
stop
else
continue
End

End

Once the corners of the eyes are detected, midpoint of the two corners gives good approximation for the center of the eyes if it is a frontal image. However, the error increases if the face is directed further away from the front. Therefore, instead of using simple midpoint, the following equations are used to find the center of the eyes.

\[
\text{Center}_{LeftEye,x} = \frac{d^2_{Left}}{1 + \left(\frac{y_{LR} - y_{LL}}{x_{LR} - x_{LL}}\right)^2} + x_{LL} \quad (1)
\]

\[
\text{Center}_{LeftEye,y} = \frac{y_{LR} - y_{LL}}{x_{LR} - x_{LL}} \ast (\text{Center}_{LeftEye,x} - x_{LL}) + y_{LL} \quad (2)
\]

\[
\text{Center}_{RightEye,x} = \frac{d^2_{Right}}{1 + \left(\frac{y_{RR} - y_{RL}}{x_{RR} - x_{RL}}\right)^2} + x_{RL} \quad (3)
\]

\[
\text{Center}_{RightEye,y} = \frac{y_{RR} - y_{RL}}{x_{RR} - x_{RL}} \ast (\text{Center}_{RightEye,x} - x_{RL}) + y_{RL} \quad (4)
\]

where \((x_{LL},y_{LL}), (x_{LR},y_{LR}), (x_{RL},y_{RL}), (x_{RR},y_{RR})\) are the corners of eyes as shown in Figure 3 and

\[
d_{Left} = \left| \frac{d_L}{d_R} \right| \ast \sqrt{\frac{(x_{LR} - x_{LL})^2 + (y_{LR} - y_{LL})^2}{2}} \quad (5)
\]

\[
d_{Right} = (1 - 0.1 \ast \frac{d_L}{d_R}) \ast \sqrt{\frac{(x_{RR} - x_{RL})^2 + (y_{RR} - y_{RL})^2}{2}} \quad (6)
\]

The above equations are used when \(\frac{d_L}{d_R}\) is less than or equal to 1. These equations are reversed, i.e. left corner coordinates are replaced by the right corner coordinates, when \(\frac{d_L}{d_R}\) is greater than 1.

4 MOUTH DETECTION

Once the centers of both eyes are known, the image is rotated so that the eyes become horizontal to each other. The lower half of the face image is kept as target for the mouth. The mouth detection algorithm presented in Hsu et al. (2002) is used to detect the mouth location. After applying the mouth map, a simple thresholding is sufficient to identify the mouth and its two corners. Again, simple midpoint is not a good approximation for the center of the
mouth. Therefore, the following equations are used to find the center of the mouth.

\[
Center_{\text{Mouth}}^x = \sqrt{\frac{d_{\text{Mouth}}^2}{1 + \left(\frac{y_{\text{MR}} - y_{\text{ML}}}{x_{\text{MR}} - x_{\text{ML}}}\right)^2}} + x_{\text{ML}} \quad (7)
\]

\[
Center_{\text{Mouth}}^y = \left(\frac{y_{\text{MR}} - y_{\text{ML}}}{x_{\text{MR}} - x_{\text{ML}}}\right) \times (Center_{\text{Mouth}}^x - x_{\text{ML}}) + y_{\text{ML}} \quad (8)
\]

Where \(x_{\text{ML}}, y_{\text{ML}}, x_{\text{MR}}\) and \(y_{\text{MR}}\) are the mouth corners as shown in Figure 5 and

\[
d_{\text{Mouth}} = \frac{d_L}{d_R} \times \sqrt{\frac{(x_{\text{MR}} - x_{\text{ML}})^2 + (y_{\text{MR}} - y_{\text{ML}})^2}{2}} \quad (9)
\]

Figure 4 shows how the mouth corners are detected.

5 POSE ESTIMATION

The triangle made by the centers of the two eyes and mouth has a distinct shape for distinct face pose. As shown in Figure 6, the shape of the triangle changes when the pose changes and becomes isosceles when the pose is straight. In order to estimate the pose from a triangular shape, the following steps are followed:

- Detect the centers of the eyes and mouth using the methods described in sections 2 and 3.
- Rotate the triangular shape so that the centers of the eyes become horizontal to each other.
- Normalize the triangle.
- Use \(d_{\text{LM}}/d_{\text{RM}}\) to estimate the pose. Where \(d_{\text{LM}}\) and \(d_{\text{RM}}\) are the distances between the left eye and the mouth, and the right eye and the mouth in the normalized triangle respectively.

Figures 6, 7 and 8 show the original triangles, rotated triangles and normalized triangles respectively for a series of images where the Yaw changes from -60 to 60 degrees with a step of 15 degrees.

6 EXPERIMENTAL RESULTS

Standard face pose databases include: CHIL-CLEAR06, CHIL-CLEAR06, IDIAP Head Pose, CMU PIE, Softopia
HOIP, CVRR-86, CVRR-363, USF HumanID, BU Face Tracking, CVRR LISAP-14, CAS-PEAL and FacePix. All these databases either contain videos or are not available publicly. Therefore, Pointing 04 face pose database, which is available publicly, is used for testing the proposed method. The Pointing 04 corpus was a part of Pointing 2004 workshop on Visual Observation of Deictic Gestures to allow for uniform evaluation of head pose estimation systems. Pointing 04 contains 15 sets of images, with each set containing 2 series of 93 images of the same person at 93 discrete poses. The pose changes from -90 to +90 both horizontally and vertically.

Due to the images are in low resolution in the Pointing 04 database, the facial features detection methods cannot be applied robustly. Therefore, the mouth and eyes centers were selected manually and then the proposed pose estimation scheme was applied. Since the proposed method is designed mainly for horizontal pose (Yaw), the images with Yaw variations -60 to 60 degrees and pitch -15 to 15 degrees were selected for testing purpose. All of these images were classified accurately. The passing criterion was the estimated pose should be within -7.5 to 7.5 degrees of target pose. Figure 9 shows the normalized triangles for a series of images from Pointing 04 Database where the Yaw changes from -45 to 45 degrees and pitch changes from -15 to 15 degrees. From the figure it is clear that the triangular shapes can be easily clustered based on the horizontal pose(Yaw) of face.

To test the equations derived in sections 3 and 4 for eyes and mouth center estimation, an in-house database which contains images of 6 people with varying horizontal poses. Figures 10 and 11 show some of the sample results from the tests. In Figure 10, the black dots show the the of the eyes detected by the simple midpoint method. The white dots show the results of the proposed scheme. In detecting the centres of the mouth and eyes only the horizontal accuracy has been tested, which is the requirement of the proposed scheme. Figures 12 and 13 show the error comparison for a sample sequence of images, which is calculated as distance between the actual centre and detected center divided by the total length of the eye or mouth.

7 CONCLUSION AND FUTURE WORK

This paper presents a novel method for face pose estimation based on the centers of the eyes and mouth. The advantages of the method are: 1) by using only the centres of the mouth and eyes as the salient features makes the algorithm computationally effective. 2) by using the centers, instead of the corners, increases the range of pose estimation.

Further work is to improve the robustness of the method, include pitch estimation and to apply the further improved method in pose tracking in videos.
REFERENCES


