

Face Pose Estimation and its Application in Video Shot Selection

Zhiguang YANG¹, Haizhou AI¹, Bo WU¹, Shihong LAO² and Lianhong CAI¹

¹Computer Science and Technology Department, Tsinghua University, Beijing, 100084, China

²Sensing Technology Laboratory, Omron Corporation

E-mail: ahz@mail.tsinghua.edu.cn

Abstract

In this paper, a face pose estimation method and its application in video shot selection for face image preprocessing is introduced. The pose estimator is learned by a boosting regression algorithm called SquareLev.R [1] that learns poses from simple Haar-type features. It consists of two tree structured subsystems for the left-right angle and up-down angle respectively. As a specific application in video based face recognition, the best shot selection problem is discussed, which results in a real-time system that can automatically select the most frontal face from a video sequence.

1. Introduction

Face pose estimation (PE) is used to predict the 3D orientation, that is the rotation-in-plane (RIP) and rotation-out-of-plane (ROP) angles, of human head. In particular, in this paper we only discuss its simplified version that corresponds to left-right angles and up-down angles. It is very important due to face pose plays an essential role in many real-life applications, such as monitoring attentiveness of drivers [2] or automating camera management [3]. In addition, many view-based approaches for face image analysis such as face recognition usually need to estimate the pose to some extent [4].

Previous works on pose estimation (PE) include PCA [5,6], ANN [7], SVMs [8,9], and Independent Subspace Analysis (ISA) [10]. In this paper, we propose a novel method to learn a pose estimator by boosting regression algorithm called SquareLev.R [1] that learns poses from simple Haar-type features [11]. It consists of two tree structured subsystems for the left-right angle and up-down angle respectively. As a specific application in video based face recognition, the best shot selection problem is discussed, which results in a real-time system that can automatically select the most frontal face from a video sequence.

Best shot selection is of important value in live video based face related processing such as face recognition, demographic classification [12], etc. The main contribution of our work is a novel pose estimation method based on boosting regression that proves to be very useful for practical applications such as best shot selection.

The rest of this paper is organized as follows: in Section 2, we discuss the problems involved in pose estimation; in Section 3, we give a brief introduction of the boosting regression algorithm, SquareLev.R; in Section 4, we introduce the Haar feature based weak learner for regression; in Section 5, we describe our pose estimation trees; in Section 6, we give our solution to the best shot selection problem and its results; and finally in Section 7, we present our conclusions.

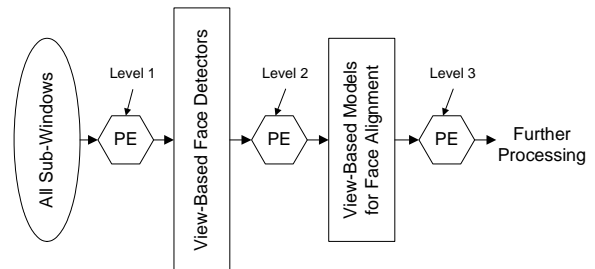


Figure 1. Definition of pose estimation (PE)

2. The Definition of Pose Estimation

As illustrated in Fig.1, PE has three variations according to its position in the flow chart. PE before face detection is a rough prediction used to divide each sub-window into its corresponding subcategory for view-based face detectors. Because there are usually millions of patches to be processed for face detection, PE at this level must be simple and fast. PE after face detection serves as the multiplexer that guides the face pattern to its view-based model. Its accuracy has direct influence on the performance of further processing. PE after face alignment is the last level. At this stage, there are usually many facial landmarks available, so

model-based method can be used. In this paper we focus on the second level. It means we assume there are no landmarks available and the target is to estimate the pose from the detected face regions in an image.

3. Boosting Regression Algorithm

The learning algorithm SquareLev.R [1] is a boost-or leverage-style regression algorithm that aims at reducing the variance of residuals. Given a sample set $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$ and a regressor F , the variance of residuals is

$$P_{Var} = \|\mathbf{r} - \bar{\mathbf{r}}\|_2^2, \quad (1)$$

where \mathbf{r} is the m -vector of residuals defined by $r_i = y_i - F(\mathbf{x}_i)$, and $\bar{\mathbf{r}}$ is the m -vector with all components equal to $\bar{r} = \frac{1}{m} \sum_{i=1}^m r_i$. Fig.2 gives the

details of SquareLev.R. It has been proved that in each iteration of SquareLev.R, P_{Var} will decrease by a factor of $(1 - \varepsilon_t^2)$ [1]. That means if ε_t has a positive lower bound ε_{min} then for any positive number ρ this algorithm will definitely generate a master regressor whose sample error is at most ρ .

- Given Sample Set $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^m$, a base learning algorithm and parameters ρ, T_{max}
- Initialize master regressor F to the zero function
- For $t = 1$ to T_{max} do
 - For $i = 1$ to m do
 - $r_i = y_i - F(\mathbf{x}_i)$
 - end do
 - If $\|\mathbf{r} - \bar{\mathbf{r}}\|_2^2 < m\rho$ break
 - For $i = 1$ to m do
 - $\tilde{y}_i = r_i - \bar{r}$
 - end do
 - Call weak learner with $S' = \{(\mathbf{x}_i, \tilde{y}_i)\}_{i=1}^m$ to obtain a hypothesis f_t
 - $$\varepsilon_t = \frac{((\mathbf{r} - \bar{\mathbf{r}}) \cdot (\mathbf{f} - \bar{\mathbf{f}}))}{\|\mathbf{r} - \bar{\mathbf{r}}\|_2 \|\mathbf{f} - \bar{\mathbf{f}}\|_2}$$
 - $$\alpha_t = \frac{\varepsilon_t \|\mathbf{r} - \bar{\mathbf{r}}\|_2}{\|\mathbf{f} - \bar{\mathbf{f}}\|_2}$$
 - $$F = F + \alpha_t f_t$$
- The final master regressor is
 - $$F(\mathbf{x}) = \sum_t \alpha_t f_t(\mathbf{x})$$

Figure 2. The SquareLev.R algorithm

4. Haar Feature Based Weak Learner

In each boosting round, SquareLev.R will call the weak learner to obtain a hypothesis or weak regressor. Different from classifications in which the hypothesis is a threshold function, the hypothesis for regression should be a continuous function of the feature value. A very simple yet effective set is the *Look-Up-Table* (LUT). We follow Viola & Jones' [11] to use the Haar features. For a Haar feature h , assuming its range has been normalized to $[0,1]$, our LUT has 64 bins and the i -th bin corresponds to the sub-domain $[(i-1)/64, i/64]$, $i=1, \dots, 64$. The hypothesis on bin $_i$ is calculated as

$$E[\tilde{y} | h(\mathbf{x}) \in \text{bin}_i]. \quad (2)$$

Define the characteristic function

$$B_i(u) = \begin{cases} 1 & u \in \text{bin}_i \\ 0 & u \notin \text{bin}_i \end{cases},$$

then the hypothesis based on Haar feature h can be formalized as

$$f(\mathbf{x}) = \sum_{i=1}^{64} B_i(h(\mathbf{x})) E[\tilde{y} | h(\mathbf{x}) \in \text{bin}_i] \quad (3)$$

We construct a hypothesis pool from all possible Haar features.

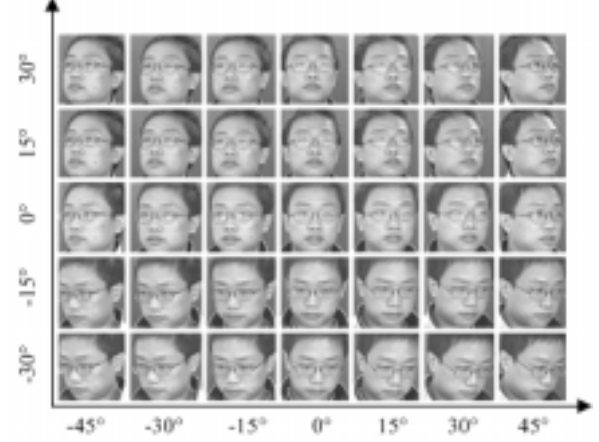


Figure 3. Multi-view face samples

5. Pose Estimation Tree

Pose data for training consist of faces with $\pm 45^\circ$, $\pm 30^\circ$, $\pm 15^\circ$, 0° left-right ROP and $\pm 30^\circ$, $\pm 15^\circ$, 0° up-down ROP that is totally 35 view categories of which each has 300 faces of different people. Because our target is PE after face detection, we do not do any shape alignment to the face samples, that is to say the face block obtained by the face detection module will

be used for training directly. All samples are resized to 24×24 -pixel patch, see Fig.3.

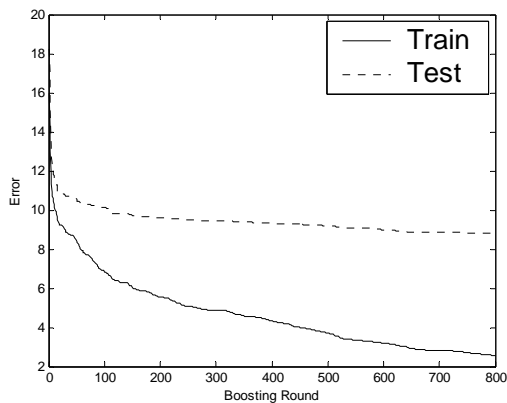


Figure 4. Error curve

The algorithm SquareLev.R is used to train left-right and up-down pose estimators. Fig.4 illustrates the error curves of the left-right regressor on the training and testing sets. It can be seen that after a few hundreds of rounds, the generalization offers minute improvement, although the sample error on training set still decrease. To overcome this disadvantage, a hierarchy structure of the so-called pose estimation tree is proposed, see Fig.5. For example, the left-right tree has two levels. The first level has one root node that covers all views. The second level has five leaf nodes that cover $[-45^\circ, -15^\circ]$, $[-30^\circ, 0^\circ]$, $[-15^\circ, 15^\circ]$, $[0^\circ, 30^\circ]$ and $[15^\circ, 45^\circ]$ respectively. For an input sample, the root only gives a rough estimation of its pose, and the more accurate estimation is left to its corresponding leaves. Since the root does not need to be highly accurate and the leaf only deals with a small sub-range, they can be implemented with relatively few features. In our experiments each leaf in the tree has 400 features, and the root has 800 features. In five-folder cross validation, the average errors of the left-right tree and up-down tree are 8.8 degrees and 9.8 degrees respectively. Table 1 lists the errors on various views and Fig.6 shows the estimators' error distributions on samples.

Table 1. The estimators' average errors on various views (LR: left-right; UD: up-down)

	-45°	-30°	-15°	0°	15°	30°	45°
LR	11.6°	9.7°	7.1°	6.3°	7.0°	8.0°	9.9°
UD	-	12.4°	7.5°	7.0°	7.4°	12.8°	-

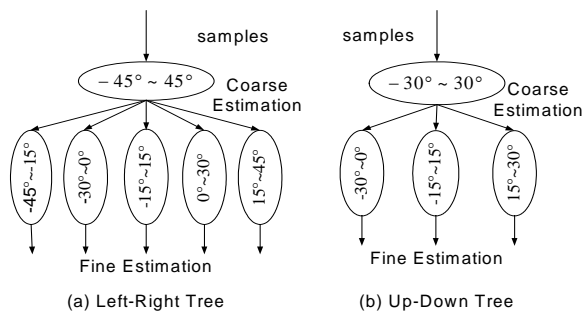


Figure 5. Pose estimation trees

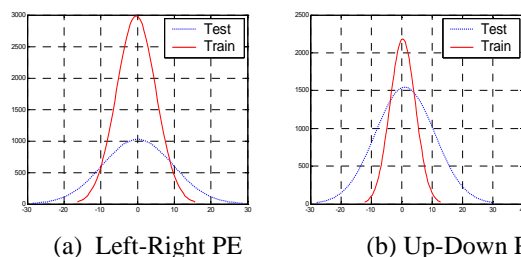


Figure 6. The estimators' error distributions on samples

6. Best Shot Selection

Based on this pose estimation method, we have developed a video-based best shot selection system of which its flowchart is given in Fig.7. First we use three cascade face detectors, a frontal detector, a left half profile detector and a right half profile detector, to detect faces in each frame. Second, the pose estimator is applied to these faces to obtain the left-right and up-down ROP angles. Finally, the face with the smallest rotation angles is selected as the best shot. Fig.8 presents an example. In best shot selection, we use two-level trees illustrated in Section 5 to estimate poses due to large ROP angle changes involved.

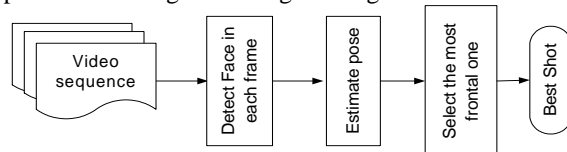


Figure 7. Flow chart of best shot selection

To test our system, we have collected three groups of video sequences, including coarse and delicate sequences. In the first group, volunteers are required to rotate their heads from left to right; in the second group, they are required to move their heads from up to down; in the third group, they are required to rotate their heads from upper left to lower right, see Figure 8 for an example. Each group has 25 coarse sequences

and 30 delicate sequences of different faces, each coarse sequence has 10 frames and each delicate sequence has 150 frames. The faces with $-10^{\circ}\sim 10^{\circ}$ left-right ROP and $-10^{\circ}\sim 10^{\circ}$ up-down ROP are accepted as frontal pose and others are rejected. Each frame is labeled to be frontal or not frontal subjectively as the ground truth data. Table 2 shows the results obtained by our system on the three groups of test data. It is quite encouraging. Those selected shots which are considered not the ground truth best shot are in fact rather close to the true one that makes our algorithm very useful in practice. The speed is about 14 frames per second (fps) on a 1.4GHz Athlon PC.

Table 2. Results of Best Shot Selection (Each group consists of 55 sequences)

Test group	Result	Percent of acceptance
left to right		81.4%
up to down		70.9%
upper-left to lower-right		79.5%

7. Conclusion

In this paper, we have applied boosting regression algorithm to learn face pose estimator. We have integrated pose estimation with face detection into an automatic real-time best shot selection system, which can be used in pose-sensitive face-related preprocessing such as face recognition. We believe the proposed method is also promising for other image regression related problems.

8. Acknowledgements

This work was supported mainly by a grant from OMRON Corporation. It was also supported in part by National Science Foundation of China under grant no.60332010.

8. References

- [1] Nigel Duffy and David P. Helmbold, "Boosting Methods for Regression", Machine Learning, Vol.47, No. 2-3, pp.153-200, 2002.
- [2] Qiang Ji and Xiaojie Yang, "Real-Time Eye, Gaze, and Face Pose Tracking for Monitoring Driver Vigilance", Real-Time Imaging 8, pp.357-377, 2002.
- [3] Qiong Liu, Yong Rui, Anoop Gupta and JJ Cadiz, "Automating Camera Management for Lecture Room Environment", In ACM CHI, 2000.
- [4] Alex Pentland, Baback Moghaddam and Thad Starner, "View-Based and Modular Eigenspaces for Face Recognition", CVPR 1994.
- [5] Shaogang Gong, Stephen McKenna and John J. Collins, "An Investigation into Face Pose Distributions", FG1996.
- [6] Mukesh C. Motwani and Qiang Ji, "3D Face Pose Discrimination using Wavelets", ICIP2001.
- [7] V. Krueger and G. Sommer, "Wavelet Networks for Face Processing", Journal of the Optical Society of America(JOSA) 2002.
- [8] Jeffrey Ng and Shaogang Gong, "Composite Support Vector Machines for Detection of Faces across Views and Pose Estimation", Image and Vision Computing 20, pp. 359-368, 2002.
- [9] Jeffrey Huang, Xuhui Shao and Harry Wechsler, "Face Pose Discrimination Using Support Vector Machines (SVM)", ICPR98.
- [10] Stan Z. Li, Xianhuan Peng, Xinwen Hou, Hongjiang Zhang, Qiansheng Cheng, "Multi-View Face Pose Estimation Based on Supervised ISA Learning", FG2002.
- [11] P. Viola, M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", CVPR2001.
- [12] G. Shakhnarovich, P. Viola and B. Moghaddam. "A Unified Learning Framework for Real Time Face Detection and Classification". FG 2002.

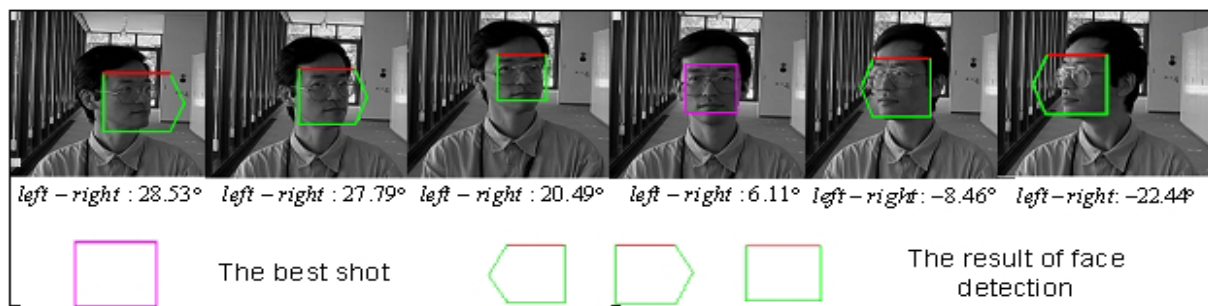


Figure 8. An example of best shot selection