

# REAL-TIME FACE ALIGNMENT WITH TRACKING IN VIDEO

Yanchao SU<sup>1</sup>, Haizhou AI<sup>1</sup>, Shihong LAO<sup>2</sup>

*1 Computer Science and Technology Department, Tsinghua University, China*

*2 Sensing and Control Technology Laboratory, Omron Corporation, Japan  
ahz@mail.tsinghua.edu.cn*

## ABSTRACT

Real-time face alignment in video is very critical in many applications such as facial expression analysis, driver fatigue monitoring, etc. This paper presents a real time algorithm for face alignment in video that combines Active Shape Model (ASM) based face alignment and spatial-temporal continuity based tracking strategy. To guarantee the correctness of the tracked shape in each frame, a verification procedure is introduced so that when inter-frame shape tracking failed the intra-frame ASM algorithm can be restored to initialize a new shape for tracking. Experiments show that the implemented system can run totally automatic with a quite good accuracy that may have many practical applications.

*Index Terms*— Face alignment, tracking, ASM

## 1. INTRODUCTION

Face alignment (FA) in images and videos is usually an essential preprocessing step of many face related computer vision tasks such as 3D face modeling; pose estimation; expression analysis and face recognition. Model based methods are widely used in FA approaches, of which Active Shape Model (ASM) and Active Appearance Model (AAM) are two classic methods [1]. In the literature, there have been many derivatives of those classical ASM and AAM methods such as Direct Appearance Model [2], Gabor wavelet [3], Haar wavelet [4], Ranking-Boost [5] and Fisher-Boost [6], etc.

Generally speaking, in both ASM and AAM methods, the shape is represented by a set of feature points that is constrained by a PCA shape model called Point Distribution Model (PDM), but their feature models are different. ASM represents the texture of each feature point by a local model, which is only related to a small local neighborhood of the feature point; while AAM has a global appearance model, in which the entire face texture is used to conduct the optimization of shape parameters. AAM is sensitive to the illumination and noisy background texture due to its global texture model. ASM performs more accurately on shape localization, and is relatively more robust to illumination and bad initialization.

After about one decade's research, FA has achieved significant progress that makes it a useful tool in face vision related researches. Although there are many FA approaches for still face images, few discussed the FA problem in video of which fast speed and robustness are two critical issues. In this paper we first extend a recent variation of ASM method [7] that uses boosted local texture classifiers as local models, and then present a FA framework that combines Active Shape Model (ASM) based face alignment and spatial-temporal continuity based tracking strategy, which results in real-time robust FA performance in video. In this framework, intra-frame FA initiates alignment tracking in successive frames while the correctness of the tracked shape in each frame is verified by a special procedure so that when inter-frame shape tracking failed the intra-frame FA can be restored to initialize next round tracking.

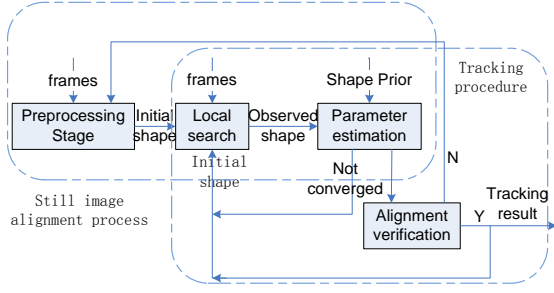
The rest of this paper is organized as follow: section 2 describes our framework in detail, section 3 gives some experiment results to show the robustness and efficiency of our framework and finally conclusion is given in section 4.

## 2. FACE SHAPE TRACKING FRAMEWORK

In face alignment, face shape is represented by a set of  $N$  feature points  $(x_i, y_i)$  concatenated as a shape vector  $s=(x_1, y_1, x_2, y_2, \dots, x_N, y_N)$ . Given a video or an image sequence  $\{I^t\}$ , the objective is to find corresponding face shape sequence  $\{S^t\}$  of all the frames. Suppose Markov condition holds, that is to say, the state (shape) of current frame depends only on its previous frame and the observation (texture) of current frame is independent of previous ones. With this assumption, the shape of frame  $t$  ( $S^t$ ) could be inferred by the shape of previous frame ( $S^{t-1}$ ) and the texture of frame  $t$  ( $I^t$ ).

The proposed framework is illustrated in Figure 1. The basic idea is to use previous result as current initialization for local search, and in order to avoid bad initialization due to failure of previous frame alignment (result) verification procedure is introduced.

In this framework, the first frame of video is aligned with a complete ASM alignment process as done in a still image,



**Figure 1. Diagram of Tracking Framework**

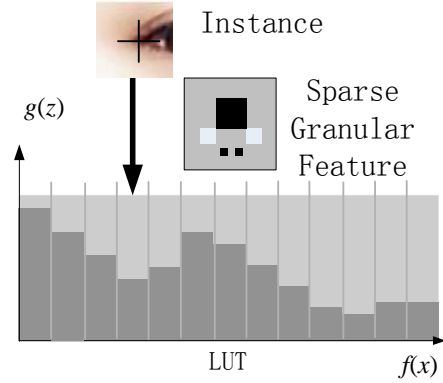
which we called as intra-frame alignment. Alignment verification step is needed to verify whether the shape is reasonable based on its embedded texture. Once we have got a reasonable aligned shape, the tracking procedure starts. The current shape is directly used as the initial shape of the next frame, and starts a local search. Also we can get a tracking prior shape by motion prediction with the previous tracking result. In the parameter estimation stage, the observed shape, the shape prior and the tracking prior are fused together to achieve the aligned shape. It is done in an iterative procedure.

In this framework, there are three key issues: local search, parameter estimation and alignment verification.

## 2.1 Local Search of Feature Points

As stated in [7], using discriminative model such as boosted classifiers as the local texture model will improve both the accuracy and the robustness of alignment algorithms due to its great discriminative power learned over large training data. In its original form, it used Haar-like feature based boosted classifiers. Recently sparse granular features [8] were proposed in face detection field which is more representative and demonstrated better performance than Haar-like features. In this paper, we use sparse granular features instead of Haar-like features to boost local texture models to enhance the local texture classifier based ASM method [7].

As shown in Figure 2, a sparse granular feature is defined as a linear combination of several granules of different positions and scales, which projects a given high dimensional instance space onto a discriminative 1D feature space that is divided into a set of bins of equal width as the partition of original instance space to construct a weak hypothesis over training data. Real AdaBoost [9] is used to learn classifiers as local texture models to discriminate feature points against non feature points with image patches centered in a given neighborhood area. The objective of local search in classical ASM methods is to find the best candidate for each feature points. In our framework, instead of searching for the best candidate of each label point, we approximate its probability distribution around that point. In practice, we sample uniformly around the initial location of each feature point and use the trained local texture classifier



**Figure 2. Illustration of a weak hypothesis. An instance is mapped onto a 1D feature space via a sparse granular feature, and the 1D feature space is further divided into a set of equal-width bins.**

to get the confidence of each sampled point. And then we approximate the probability distributions as Gaussian models:

$$P(I | x_i) \sim N(x_i^*, \Sigma_i) \quad (1)$$

Where  $\{x_i^*, \Sigma_i\}$  indicate the observed shape and its probability characteristic of each feature point.

These distributions will be used in the parameter estimation step.

## 2.2 Fusion of Observed Shape and Prior Shapes

Alignment in video aims to find the shape of each face according to the embedded texture in each frame, that is, given a series of frames  $\{I^1, I^2, \dots, I^N\}$ , find the best shape series  $\{x^1, x^2, \dots, x^N\}$  to maximize the probability of  $P(x^1, x^2, \dots, x^N | I^1, I^2, \dots, I^N)$

To reduce the complexity of the problem, we can suppose that Markov condition holds. So when we are aligning the  $i$ -th frame:

$$\begin{aligned} x^i &= \arg \max P(x^1, x^2, \dots, x^i | I^1, I^2, \dots, I^i) \\ &= \arg \max P(x^i | I^i, x^{i-1}) \\ &= \arg \max P(x^i | I^i) P(x^i | x^{i-1}) \end{aligned} \quad (2)$$

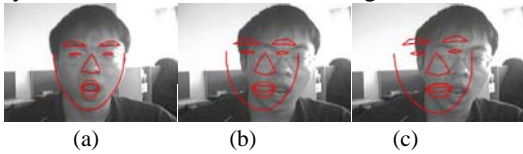
By applying Bayesian rule on equation 2, we can get:

$$\begin{aligned} x^i &= \arg \max P(x^i | I^i) P(x^i | x^{i-1}) \\ &= \arg \max P(x^i | x^{i-1}) P(I^i | x^i) P(x^i) \end{aligned} \quad (3)$$

In equation 3, three different models are fused together to infer the current shape. In the equation,  $P(x^i | x^{i-1})$  indicates how the tracked shape will change between the frames; and  $P(I^i | x^i)$  is the probability of the texture around feature points according to the local texture model. And  $P(x^i)$  is the prior probability of the shape. This optimization can be solved using Gaussian Newton method.

### 2.3 Verification of Shapes

When the iteration of aligning a frame converges, we get an alignment result of current frame. In our framework of tracking, this result is used to predict the initial shape of the next frame. So ensuring the correctness of the alignment result of each frame is very important for initiating the next frame. Once a wrong alignment resulted in some frame, the initial shape of its predicted next would be far from the ground truth of the frame. In this case, the alignment procedure in this frame would take more time to process or even fail to converge. To the worse, that kind of error will be spread to the succeeding frames and makes the shape drift away from the face as illustrated in Figure 3.



**Figure 3. Alignment in video without verification**

(a). Aligned correctly in this frame; (b). Failed due to quick movement of the face; (c) Without verification, the alignment failed to converge and drifted away due to bad initialization predicted from (b).

A direct way of verifying the alignment result is by means of the texture embedded in the shape. To eliminate the texture variations caused by expression and pose changes, we warp the embedded texture into a reference shape, usually the mean shape of PCA model, to get the shape free texture of the faces. It is not difficult to understand that bad alignment results will produce unreasonable face textures due to non-face patterns the shape embedded, see Figure 4 for example. By verifying the shape free texture we could get a confidence value of each alignment result.



**Figure 4. Samples of shape free textures on images from FERET [12]** (a),(b): correct alignment result and its shape free textures; (c),(d): wrong alignment result and its shape free textures

A common and simple way to verify alignment result is using the reconstruction error used in AAM. That is, modeling the positive samples of shape free texture using PCA method, and using the reconstruction error of PCA model to verify the alignment results. This method uses generative model to characterize the positive samples of shape free texture and thus finds out the wrong ones.

An alternative method to solve this problem is trying to distinguish the negative samples from the positive ones, that

is, using a discriminative model such as Linear Discriminative Analysis (LDA).

In our experiments, LDA method performs much better than simply using the reconstruction error of PCA model. But both of them use the global appearance and thus suffer from the illumination variation and the change of datasets, as it is in AAM alignment. Experiment shows that LDA is much better than PCA but are too easy to get over fit to the training set.

Another method to verify alignment result is to train a cascade classifier to discriminate positive result against negative ones [10][11]. And experiment result shows that boosted classifier using Haar-like feature performs more robust than the simple linear methods such as PCA error and LDA on different datasets.

### 3. EXPERIMENTS

We have tested our algorithm on a PC with 3GHz CPU and 2GB RAM, on a couple of videos captured by web camera and some sequences from TV shows. We observed that the algorithms cost about 54ms with about 3 iterations at average to process each frame. On a video with 301 frames, the inter-frame alignment failed 4 times and the failures were detected correctly by the alignment verification step. For illustration, see Figure 5, and some results are shown in Figure 6.

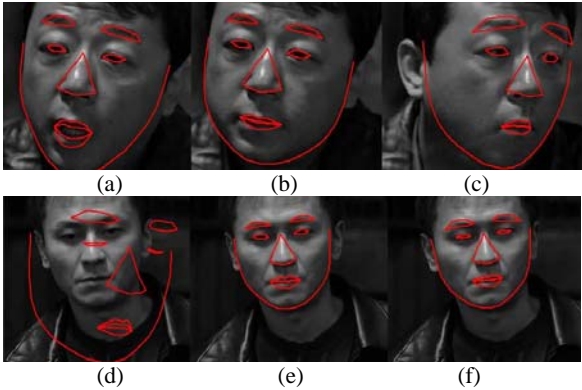
To show the speed and the robustness of our framework, we compares our algorithm with classic Active Shape Model with local texture classifier which process video frame-by-frame. The experiment was carried on the same video sequence with 301frames. Results are shown in Table 1. The process time includes the time cost by face detection.

**Table 1. Comparison between our algorithm and the frame-by-frame ASM**

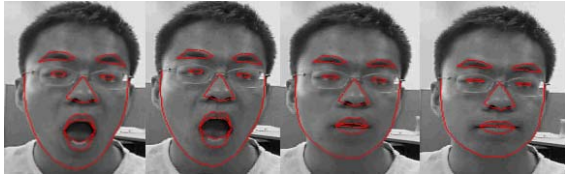
|                | Ours  | frame-by-frame |
|----------------|-------|----------------|
| Process time   | 54ms  | 93ms           |
| Average errors | 2.1px | 2.2px          |

Since the other sequences are clipped from TV shows and movies, we don't have the ground truth labels of the faces in these videos. So we can't give the precise statistics of the performance of the method. On the 1034 frames of all the sequences, the inter-frame alignment failed 13 times, including 5 times that were caused by the change of scenes, and the failures were all correctly detected. And the alignment results are acceptable over all the frames.

In the video sequence, there are some frames with bad illumination that caused miss-detection of faces and therefore the alignment failed in frame-by-frame ASM method, while in our tracking algorithm, shapes are initiated from the previous frame so these frames can be processed correctly.



**Figure 5. Illustration of the tracking framework** ((a), (b), (c): Alignment goes smoothly through the frames over 70 frames; (d): When the camera-shot changes, resulted in a bad alignment result which is detected by the verification module and (e) the intra-frame alignment is restarted to get reasonable result, then (f) the tracking goes smoothly as before)



**Figure 6. Tracking Alignment result in videos**

#### 4. CONCLUSION

In this paper, we propose a face shape tracking framework with alignment result verification. We take advantage of the spatial-temporal continuity of videos to achieve faster speed and more accuracy. Based on the classic ASM method, boosted sparse-feature classifiers are used to distinguish the feature points from its neighborhoods to obtain more accurate local search result. We model the local likelihood distribution of each feature points to get more information of local textures. And at the parameter estimation step, multiple cues, such as shape model, local likelihood distribution of feature points and tracking prediction, are fused together in a probability framework to form the final shape. Also we adopt an alignment result verification step to guarantee the correctness of the alignment in each frame and restart the intra-frame alignment when failure is detected.

Face alignment with tracking makes use of the continuity of video, so it takes less iteration to converge on each frame

and is much faster than doing alignment each frame independently. Also, alignment with tracking avoids detecting face in every frame so that frames in bad condition could be processed correctly even if the face could not be detected, which makes the alignment result more stable over the whole video. Experiment shows that our algorithms could process video in real-time.

#### 5. ACKNOWLEDGEMENT

This work is supported in part by National Science Foundation of China under grant No.60673107, and it is also supported by a grant from Omron Corporation.

#### 6. REFERENCES

- [1] T. F. Cootes, Statistical models of appearance for computer vision, <http://www.isbe.man.ac.uk/~bim/refs.html>, Sept. 2001.
- [2] S. Z. Li, S.C Yan, et.al, Multi-view face alignment using direct appearance models, AFG 2002.
- [3] F. Jiao, S. Z. Li, et.al, Dale Schuurmans, Face alignment using statistical models and wavelet features, CVPR 2003.
- [4] F. Zuo, Peter H.N. de With, Fast facial feature extraction using a deformable shape model with Haar-wavelet based local texture attributes, ICIP 2004.
- [5] S. Yan, M. Li, et.al, Ranking prior likelihood distributions for Bayesian shape localization framework, ICCV 2003.
- [6] J. Tu, Z. Zhang, et.al, Face localization via hierarchical CONDENSATION with Fisher Boosting feature selection, CVPR 2004.
- [7] L. Zhang, H. Ai, et al., Robust Face Alignment Based on Local Texture Classifiers, ICIP 2005.
- [8] C. Huang, H. Ai, et.al, Learning Sparse Features in Granular Space for Multi-View Face Detection, In Proc. AFG 2006.
- [9] R. E. Schapire and Y. Singer, Improved Boosting Algorithms Using Confidence-rated Predictions, Machine Learning, 37, 1999, pp. 297-336.
- [10] P. Viola and M. Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, CVPR 2001.
- [11] C. Huang, H. Ai, et.al, Boosting Nested Cascade Detector for Multi-View Face Detection, ICPR 2004.
- [12] P.J. Phillips, et.al, The FERET database and evaluation procedure for face recognition algorithms. Image and Vision Computing J (1998)