11-2005

Exploiting self-adaptive posture-based focus estimation for lecture video editing

Feng WANG

Chong-wah NGO
Singapore Management University, cwngo@smu.edu.sg

Ting-Chuen PONG

Exploiting Self-Adaptive Posture-based Focus Estimation for Lecture Video Editing

Feng Wang
Dept. of Computer Science
Hong Kong University of Science and Technology
wfeng@cs.ust.hk

Chong-Wah Ngo
Dept. of Computer Science
City University of Hong Kong
cwngo@cs.cityu.edu.hk

Ting-Chuen Pong
Dept. of Computer Science
Hong Kong University of Science and Technology
tcpong@cs.ust.hk

ABSTRACT

Head pose plays a special role in estimating a presenter's focuses and actions for lecture video editing. This paper presents an efficient and robust head pose estimation algorithm to cope with the new challenges arising in the content management of lecture videos. These challenges include speed requirement, low video quality, variant presenting styles and complex settings in modern classrooms. Our algorithm is based on a robust hierarchical representation of skin color clustering and a set of pose templates that are automatically trained. Contextual information is also considered to refine pose estimation. Most importantly, we propose an online learning approach to deal with different presenting styles, which has not been addressed before. We show that the proposed approach can significantly improve the performance of pose estimation. In addition, we also describe how posture is used in focus estimation for lecture video editing by integrating with gesture.

Categories and Subject Descriptors: K.3.1 [Computer Uses in Education]: Distance learning. **General Terms:** Algorithms, Design, Experimentation. **Keywords:** Pose Estimation, Lecture Video, Video Editing.

1. INTRODUCTION

Due to the popularity of e-learning, lecture videos are widely available for online access. To support effective browsing and search, these videos need to be properly captured, indexed and edited. Traditionally, this work is mostly operated by expert cameramen and editors. This procedure, nevertheless, is costly and requires manual work. Recently, numerous attempts have been made to automate the multimedia authoring of live presentations. These efforts include off-line video editing [1] and real-time broadcast by automatic camera management [3, 5]. One desired goal is to correctly predict what presenters want to highlight, at any moment, and produce videos with appropriate views. This goal requires effective recognition of video content such as the speech, gesture and posture of a presenter.

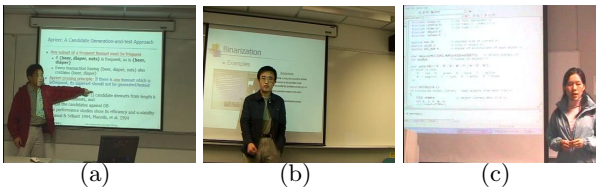
Presentation capture has imposed several new technical challenges for face detection, tracking and pose recognition. The difficulties are mainly due to the complex lighting conditions in classrooms and the low resolution of faces in videos. Fig 1 shows some examples. In (a), the front lights are turned off in order to make the text on the screen visible. When the presenter stands in front of the screen, half of the face looks dark while the other half is illuminated by the light from the LCD projector. In (b), the face is overlaid with the slide image emitted from the projector. In (c), the face has the similar color with the background, which makes the detection difficult. For most lecture videos that we have collected, the area of a face occupies only approximately 1% of a video frame.

This paper presents an efficient head pose estimation approach, suitable for both off-line and real-time video editing, to effectively estimate the *focus of lecturing*. To deal with complex lighting environments, we first propose a hierarchical representation for robust skin color detection and clustering. A set of pose templates are automatically trained for rapid pose estimation.

Besides visual cues, we focus more on effectively exploiting contextual information, *i.e.* temporal smoothness of head movement to refine the pose estimation, which is useful especially for low-resolution images where direct estimation from one single image is not reliable enough. Although contextual information has been used in pose estimation, different from most other works with fixed contextual information, we propose an adaptive learning approach to cope with the pace of different presenting styles. The presenting style is represented by a set of pose transition probabilities, which can be adapted to different lecturers by online learning. As we know, this is the first work that addresses the effect brought by different lecturers and presenting styles in pose estimation. Experiments show that the overall estimation performance is significantly improved. The proposed approach has potential use for pose estimation in other videos, and for other human activity recognition problems. Finally we describe how the head pose is used for lecture video editing to estimate the focus of lecturing.

Recent works related to our approach include [3, 5]. Both approaches aim for real-time camera management. In [5], basic techniques such as frame difference are used and thus only simple gestures can be recognized to detect the focus of lecturing. In [3], head pose is simply estimated based on the

MM'05, November 6–11, 2005, Singapore.



(a) (b) (c)

Figure 1: Challenges of pose estimation in lecture videos

number of skin color pixels lying on each side of a face. Most existing approaches including [3, 5] do not address the issues of lighting conditions particularly for classrooms equipped with LCD projected screens and when a presenter is allowed to interact with the screen.

2. VIDEO PREPROCESSING

In this section, we describe our algorithm for robust face detection and tracking. Skin color has been shown to be a reliable cue for face detection in videos and color images [2]. To rapidly locate candidate faces, we adopt a rule-based classifier in [4] to efficiently detect skin color pixels. The detection is carried out in a multi-resolution manner. Potential pixels are initially detected at down-sampled frames and subsequently refined in higher resolution frames. The classified skin color pixels are then clustered for face detection.

Skin color detection is vulnerable to noise due to lighting conditions and colors that are similar to skin. Our skin color classifier indeed occupies a rather large region in color space by considering different races and brightness. Most skin pixels can be correctly detected if the color of the face is not changed significantly when being projected by LCD light. Some noises, nevertheless, will be falsely included which can ultimately affect the clustering of skin pixels.

Fig 2 shows two types of noise that are difficult to deal with in lecture videos. In (a), the face appears to be split into two with different illuminations. In (b), the shirt has skin-like color. To robustly handle noise, we propose a two-level hierarchical clustering algorithm. Skin density is considered at the first level while the difference of skin color is utilized for further decomposition at the second level. Initially, all the detected skin color pixels are grouped into a set of blocks S_1 based on density clustering. Each block in S_1 is then segmented into several clusters according to color difference. The blocks in both levels will be further processed in candidate face selection. Fig 2 shows the results of hierarchical clustering. In (a), the face remains as a whole at the top level although it has been segmented into two parts at the lower level. In (b), the face and shirt are decomposed as two parts in the second level by color difference.

To detect frontal faces, we fit an ellipse for each skin cluster and then select the candidate faces. Fig 2 shows two faces that are fitted by ellipses. After fitting, by considering the camera setting and the general shape of human face, some clusters are excluded based on ellipse size, ratio of height and width, skin color density and color variance. The frontal pose templates $T^m_{\rho 3}$ (described in Section 3 and 4) are further employed to confirm the existence of faces in candidate clusters. Once a frontal face is identified, it is continuously tracked over frames. For speed efficiency, face tracking is constrained in a smaller region near the previous

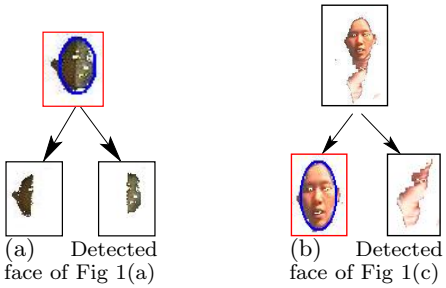


Figure 2: Hierachical clustering of skin color pixels

detected position. Color, size and location are taken into account when selecting the most similar face for tracking.

3. POSE TEMPLATE GENERATION

Conventional approaches to pose estimation are usually composed of two parts: feature extraction and classification. The former can be a time consuming process particularly if complex features are extracted. To be more efficient, we extract features during training, and omit online feature extraction. Basically, a set of face templates are automatically trained and generated as pose representatives for estimation.

In this paper, we only consider poses along x-axis direction. These poses usually provide more useful hints for lecture video editing. Notice that the precise estimation of each pose (in terms of degree of orientation) is indeed not necessary since our ultimate task is to estimate the focus of lecturing. We quantize the orientation of a face into five different poses $\rho_1 \dots \rho_5$, from 15^o to 165^o, separated by 30^o. The poses ρ_1 and ρ_5 denote facing the extreme left and right directions respectively. We collect five set of training faces, one for each pose, from various people. During training, each face $\mathcal{F}$ is scaled to 48×64. A gradient image G_i is computed and compensated for the unevenness and variety of illumination. The histogram of G_i is then normalized in the range $[0 - 255]$ to sharpen the image contrast. For each training set P_i, we compute an average pose-dependent face

$$\Upsilon_i(j,k) = \frac{1}{n_i} \sum_{\mathcal{F} \in P_i} \mathcal{F}(j,k) \tag{1}$$

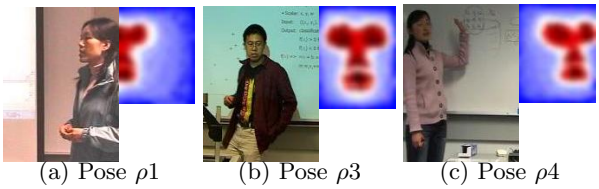
where (j,k) indexes the pixels in $\mathcal{F}$ and Υ_i. The average face across N different poses is computed as

$$\Gamma(j,k) = \frac{1}{N} \sum_{i=1}^{N} \Upsilon_i(j,k) \tag{2}$$

The pose templates are generated by comparing the average pose-dependent faces. The template of a pose ρ_i is computed as

$$\Delta_{\rho_i}(j,k) = \begin{cases} \frac{\Upsilon_i(j,k) - \Gamma(j,k)}{\Gamma(j,k)} & \text{if } \Upsilon_i(j,k) > \alpha\Gamma(j,k) \\ & \text{or } \Upsilon_i(j,k) < \beta\Gamma(j,k) \\ 0 & \text{otherwise} \end{cases}$$

Two parameters α and β are empirically trained ($\alpha = 1.3$ and $\beta = 0.7$). A pose template is characterized by the features inherent in Δ_{ρ_i}. These features are the points whose values are significantly higher or lower than others. Fig 3 shows three different pose templates. Salient facial features are occupied by points with high and positive values (red



(a) Pose $\rho 1$ (b) Pose $\rho 3$ (c) Pose $\rho 4$

Figure 3: Different poses and their templates

and black regions in Fig 3), while the regions without facial features are filled with low and negative values (white and blue regions). To tolerate variations caused by face shape, gender and race, we train K sets of pose representatives. Currently, we set $K = 4$ (two sets for female and another two for male), based on the gender and ratio of height and width. Each set is composed of five pose templates.

4. DIRECT POSE ESTIMATION

The generated templates are used for pose estimate by calculating their fitness with the detected and tracked faces. The face $\mathcal{F}$ is first scaled and normalized as in template generation. An ellipse is fitted and the face is rotated to make the longer axis of the ellipse along the vertical direction. The fitness of $\mathcal{F}$ with a template $T^m_{\rho_i}$ of pose ρ_i in template set m is computed as

$$\mathcal{C}(\mathcal{F}, T^m_{\rho_i}) = \max_{-1\le a,b\le 1} \sum_{j,k} \mathcal{F}(j+5a, k+5b) \cdot T^m_{\rho_i}(j,k)$$

where $C(\mathcal{F}, T^m_{\rho_i})$ represents the confidence that $\mathcal{F}$ is in the mth template set with a pose ρ_i. The parameters a and b allow $\mathcal{F}$ to shift and align with the template.

In face detection, the templates $T^m_{\rho_3}$, $1 \le m \le K$, are utilized to detect frontal faces $\mathcal{F}$, where ρ_3 represents the frontal pose. The confidence of detection $\mathcal{C}_{\mathcal{M}}$ and the set of pose templates $\hat{m}$ that best fit $\mathcal{F}$ are determined as

$$\begin{aligned} \mathcal{C}_{\mathcal{M}} &= \max_{m=1}^{K} \mathcal{C}(\mathcal{F}, T^m_{\rho_3}) \\ \hat{m} &= \arg\max_{m=1}^{K} \mathcal{C}(\mathcal{F}, T^m_{\rho_3}) \end{aligned}$$

The value of $\mathcal{C}_{\mathcal{M}}$ can indicate the presence or absence of salient facial features in $\mathcal{F}$. The $\hat{m}$ actually decides the set of templates that best fit $\mathcal{F}$ for face tracking and pose recognition. Once the value of $\hat{m}$ is fixed, five pose templates $T^{\hat{m}}_{\rho_i}$, $1 \le i \le 5$, are directly used to fit and estimate the pose of the tracked face in the remaining frames.

5. SELF-ADAPTIVE ESTIMATION

With the low resolution and complex lighting conditions, the direct pose estimation from one single image is usually not reliable enough. We exploit the temporal smoothness of head movement to refine and improve pose estimation. The probability of a pose at frame $t+1$ can be inferred by the pose at frame t and the head movement between frames $t-1$ and t. By maximizing the confidence of pose transitions over frames, the state or pose s_{t+1} at frame $t+1$ is estimated as

$$\mathbf{C} = \mathcal{C}(\mathcal{F}_{t+1}, s_{t+1}) \prod_{k=t-7}^{t} \mathcal{C}(\mathcal{F}_k, s_k) p(s_{k+1}/s_k s_{k-1})$$

where $\mathcal{C}(\mathcal{F}_k, s_k)$ is the confidence value of a face $\mathcal{F}_k$ in frame k with pose s_k, and $p(s_{k+1}/s_k s_{k-1})$ is the probability of

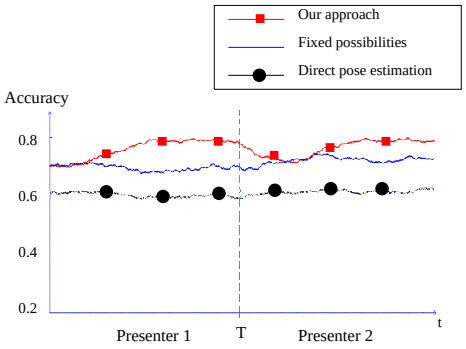


Figure 4: Pose estimation by different approaches

pose s_{k+1} in frame $k+1$ when the two previous poses are s_{k-1} and s_k. We use eight frames to estimate $\mathbf{C}$. When estimating the pose at $t+1$, we use the confidence values, rather than the recognized poses, of previous frames. This is to avoid accumulating pose estimation errors over frames.

The set of probabilities $\{p(s_{k+1}/s_k s_{k-1})\}$ can be estimated from training sets. This whole process can also be done by an HMM. However, a problem with this approach is that people tend to have different presenting styles and habits which can result in different probabilities of pose transitions. For example, some people like to talk and face audiences directly while the others may look at the screen most of the time. A fixed set of probabilities for all possible presenting styles will definitely affect the performance of pose recognition. To tackle this problem, we use self-adaptive probabilities for pose recognition. A probability set is estimated from the training videos and initialized at the beginning. When the pose at frame $t+1$ is recognized as s_{t+1}, the probability $p(s_{t+1}/s_{t-1}s_t)$ is updated as

$$p'(s_{t+1}/s_{t-1}s_t) = p(s_{t+1}/s_{t-1}s_t)(1+\delta)$$

where δ is a small value added to increase the confidence of transition probability. For any other pose $\rho \neq s_{t+1}$, $p(\rho/s_{t-1}s_t)$ is reduced accordingly by δ. The set of transition probabilities is incrementally changed until it is tailored to a specific presenting style. Usually, the probabilities will keep updated and become stable after a short period.

Figure 4 compares the performance of the three different approaches discussed for probability settings through a half-hour video, which includes two different presenters. The overall performance of adaptive probability approach is improved compared with the other two approaches. The curve also depicts the learning process of the self-adaptive probability approach. When a new presenter begins presenting, the original probability set is not appropriate again which leads to a decline of pose estimation accuracy. At the same time, a learning process starts to adapt the probability set to the new presenter. After a period of learning, the estimation accuracy is increased again. This process shows that the presenting styles do affect the pose estimation, while the pose transition probability set can reflect and utilize this effect to improve the estimation performance.

6. EXPERIMENTS

We conduct experiments on 5-hour videos consisting of 15 presentations given by 10 lecturers and tutors. The pre-

senters include 5 male and 5 female. The presentations are given in the classrooms and seminar rooms of different size, layout and lighting design. Basically two overview cameras are stationarily mounted. One captures the LCD projected screen and the other points toward a whiteboard.

6.1 Pose Estimation

Experimental results indicate that approximately 92% of the faces in videos can be correctly detected and tracked. Some errors occur when the faces are occluded or projected by the slide images of dark color from the LCD projector. Table 1 shows the results of pose estimation. We compare three approaches described in Section 4 and 5: direct estimation, estimation with fixed probability, and with self-adaptive probability. Overall, when the pose transition probabilities are introduced, the accuracy is significantly improved, especially for poses ρ_1 and ρ_5, when faces turn in the extreme left or right direction. When the self-adaptive setting is used, the performance is further improved since the transition probabilities of different presenters can be dynamically updated.

Table 1: Results of Pose Estimation

Poses	ρ_1	ρ_2	ρ_3	ρ_4	ρ_5
Direct Estimation	54%	69%	72%	66%	56%
Fixed Prob.	67%	79%	81%	75%	68%
Adaptive Prob.	75%	85%	88%	83%	76%

6.2 The Role of Pose in Video Editing

Pose is a necessary but not sufficient hint for estimating the focus of lecturing. In general, poses are useful when gestures are occluded or absent, while gestures are useful when poses are ambiguous (*e.g.*, ρ_2, ρ_4) or not seen. We conduct experiments to recognize simple actions like "facing audience", "facing screen" and "facing whiteboard" with poses and gestures in [6]. Table 2 shows the empirical results on the tested videos.

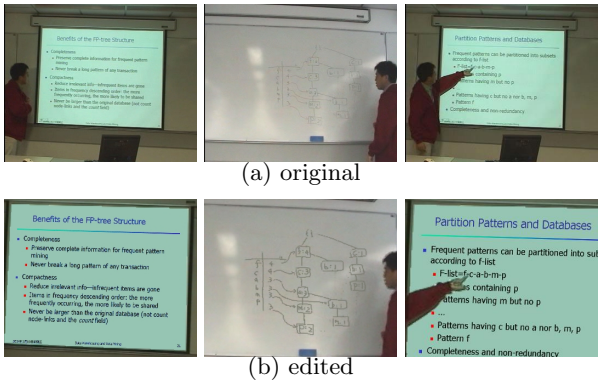
Table 2: Results of Focus Estimation

Focus	Audience	Screen	Whiteboard
Gesture	-	62%	71%
Pose	88%	74%	67%
Pose + Gesture	88%	91%	95%

When pose is combined with gesture, indeed we can estimate not only simple actions, but also synchronize the actions, for instance, with the textual content of slides. By knowing the regions of focus, we can simulate appropriate camera motions such as zoom to edit and drive the pace of lecturing. We construct a finite state machine (FSM) with 14 states which takes poses and gestures as input to edit videos with appropriate camera motion, focal length and cutting effects. When simulating zoom, we use the approach in [6] to automatically enhance the visual quality of slide and whiteboard images with symbolic documents. Fig 5 compares the original and edited frames.

6.3 Speed Efficiency

Currently, our face-gesture detector and tracker can run at 21 frames per second (fps) and pose estimation runs at 6 fps on a Pentium-IV machine. By observing the tested videos, we find that a presenter normally needs, on average, $6-15$ frames or $0.25-0.6$ second to complete a pose transition. Thus, it is not necessary to estimate a presenter's pose in every frame. Our pose estimation algorithm, although not in video rate, is actually enough not only for off-line editing, but also real-time camera management for presentation capture.



(a) original

(b) edited

Figure 5: Video editing based on focus estimation

7. CONCLUSION

We have presented an efficient approach for head pose estimation in lecture videos. Besides a hierarchical representation for robust skin color clustering and a set of pose templates automatically trained, our contribution is mainly due to the online learning method to exploit the temporal smoothness of head movement and deal with different presenting styles. Similar approaches can be used for pose estimation in other video domains and for other human activity recognition problems. With the experiment, we have empirically verified the performance of the proposed approach and demonstrated the role of pose estimation in video editing.

Acknowledgements

The work described in this paper was partially supported by the grants SSRI99/00.EG11, DAG01/02.EG16, HIA01/02.EG04 and a grant from City University of Hong Kong (Project No. 7001546).

8. REFERENCES

[1] M. Gleicher and J. Masanz, "Towards Virtual Videography", *ACM Multimedia Conf.*, 2000.

[2] R. L. Hsu, M. Abdel-Mottaleb & A. Jain, "Face Detection in Color Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 696-706, May, 2002.

[3] Masaki Onishi, Kunio Fukunaga, "Shooting the Lecture Scene Using Computer-controlled Cameras Based on Situation Understanding and Evaluation of Video Images," *Int. Conf. on Pattern Recognition*, 2004.

[4] P. Peer, J. Kovac & F. Solina, "Human skin colour clustering for face detection," *Int. Conf. on Computer as a Tool*, 2003.

[5] Y. Rui, A. Gupta & J. Grudin,"Videography for Telepresentations", *Int. Conf. on Human Factors in Computing Systems*, 2003.

[6] F. Wang, C. W. Ngo & T. C. Pong, "Gesture Tracking and Recognition for Lecture Video Editing," *Int. Conf. on Pattern Recognition*, 2004.