Estimating the Lecturer's Head Pose in Seminar Scenarios - A Multi-view Approach

Michael Voit, Kai Nickel, and Rainer Stiefelhagen

Interactive Systems Lab, Universität Karlsruhe (TH), Germany
{voit, nickel, stiefel}@ira.uka.de

Abstract. In this paper, we present a system to track the horizontal head orientation of a lecturer in a smart seminar room, which is equipped with several cameras. We automatically detect and track the face of the lecturer and use neural networks to classify his or her face orientation in each camera view. By combining the single estimates of the speaker's head orientation from multiple cameras into one joint hypothesis, we improve overall head pose estimation accuracy. We conducted experiments on annotated recordings from real seminars. Using the proposed fully automatic system we are able to correctly determine the lecturer's head pose in 59% of the time and for 8 orientation classes. In 92% of the time, the correct pose class or a neighbouring pose class (i.e. a 45 degree error) were estimated.

1 Introduction

In recent years there has been much research effort spent on building smart perceptive environments. The challenge is to build environments which support humans during their activities without obliging them to concentrate on operating complicated technical devices.

In the framework of the European Union Research project CHIL, we are therefore developing services that aim at proactively assisting people during their daily activities and in particular during their interaction with others. Here, we focus on office and lecture scenarios, as they provide a wide range of useful applications for computerized support.

To provide intelligent services in a smart lecture environment it is necessary to acquire basic information about the room, the people in it and their interactions. This includes for example the number of people, their identities, location, posture, body and head orientation, speech etc.

A person's head orientation can be a valuable cue to determine his or her focus of attention and interaction partners. This could be useful to index seminar recordings, to detect context switches such as interruptions, discussions, etc. and in particular to tell the "smart room" about the lecturer's target of attention, for instance the audience, a whiteboard, his or her laptop etc.

In this work, we present a fully automated system for tracking a lecturer's head pose. By using multiple cameras we cover the entire room and are able to combine head pose estimates coming from various camera views into one single, more robust hypothesis. To estimate head pose in each view, we use an

S. Renals and S. Bengio (Eds.): MLMI 2005, LNCS 3869, pp. 230–240, 2006.

appearance-based approach as proposed in [8], as it has proven to provide useful results even from low-resolution facial images such as the ones captured with the smart room cameras.

The remainder of this paper is organized as follows: In Section 1.1 we discuss related work. Section 2 introduces the Sensor setup used in our smart room. Section 3 gives a system overview and describes the technical components – head detection and extraction, frontal-face classification, head pose estimation and fusion – in detail. Section 4 presents experimental results on recorded seminars. In Section 5 we conclude this paper.

1.1 Related Work

In recent years, various approaches for visually estimating head pose were presented. Yet, the interacting person whose pose was to be recognized often had to limit its movement and rotation to a fixed area around the camera. This prohibits natural behaviour and only allows to embed those systems in environments where the user's freedom of movement is restricted anyway (like in a car or in front of a screen).

Especially model-based approaches as presented in [3], [2], [7] are affected by this constraint. Since in these approaches, a number of facial features need to be detected to compute head pose, they require facial images of quite high resolution and also suffer of tracking problems due to fast head movements.

In contrast, appearance-based approaches tend to achieve satisfactory results even with lower resolutions of extracted head images. In [8] a neural-network-based approach was demonstrated for head pose estimation from very low resolution facial images which were captured by a panoramic camera. Here, however, the output only covered ranges from the left to the right profile. Also only one camera view was used, thereby limiting the application of the system to an area around a meeting table.

Another interesting work is described by Ba and Obodez in [1]. They classify facial images by modelling the responses of Gabor and Gaussian filters for a number of pose classes. An interesting contribution of their work is the combination of head detection and pose estimation in one particle filter framework. However, their work was limited to a monocular system.

Tian et al. [9] described the use of wide baseline overhead stereo-cameras in a room to classify an observed head pose into one of a fixed set of discrete pose classes. Neural networks were implemented for estimating the head pose seen by each camera. A maximum-likelihood search results in the final pose hypothesis. Though the architecture of the presented system seems to be usable for more than two cameras, the work lacks an example with more than one camera pair. To our knowledge, this is the only work combining multiple views for head pose estimation.

2 Sensor Setup

Figure 1 depicts our sensor setup: four calibrated colour cameras (Sony DFW-500) are mounted in the upper corners of the smart-room at a height of about

$2.7m$. The size of the room is $5.9m \times 7.1m$. Because of this layout, the entire room is covered by the cameras' field of view, such that at least one facial view of the user's head can always be obtained. The missing ability to zoom optically results in very low-resolution images of the extracted head, depending on where the person is standing. Using the native camera resolution of 640×480 pixels, the typical size of a head is about 20×30 to 50×65 pixels in our data. Figure 4 shows the four views of the room as seen by the cameras.

In addition to the low resolution facial views, our recordings also suffer from non-optimal lighting conditions due to the non-uniform illumination coming from different light sources in the room (halogene lamps as well as sunlight coming through the windows). In our recordings, therefore, mostly two camera views are always confronted with strong back light.

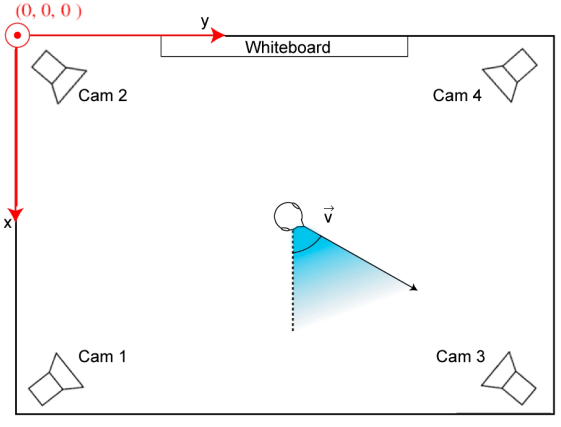

Fig. 1. Four cameras are placed in the upper corners of the smart-room, such that at least one facial view of the head can be obtained. We estimate the horizontal rotation angle (pan) of a person's head by combining the estimates from multiple cameras. The overal pose estimation is relative to the room coordinate system, situated in the north-western corner of the room.

3 System Overview

Our system for tracking a lecturer's head pose consists of the following main components:

1. Tracking of the lecturer
2. Head detection and alignment
3. Classifying frontal views vs. views at the head's back
4. Pose estimation for each camera view
5. Building a joint pose hypothesis

The following sections describe these components in detail.

3.1 Tracking the Lecturer's Head

In order to track the location of the lecturer's head, we follow the approach presented in [5]: A particle filter framework integrates multiple cues from all of the camera views and hypothesizes the lecturer's 3D position. It does so by performing sampled projections of 3D hypotheses and scoring them, thus avoiding the need for explicit triangulation.

Intuitively, the lecturer is the person that is standing and moving most, while people from the audience are generally sitting and not moving much. In order to exploit this behavior, we decided to use dynamic foreground segmentation based on adaptive background modeling [6] as primary feature. To support the track, we use detectors for face and upper body [10] [4].

In order to evaluate a hypothesis, we project a person-sized cuboid centered around the head position to the image plane, and count the number of foreground pixels inside the projected polygon. The fraction of foreground pixels within the polygon is then used as the particle's score. This calculation is sped-up by first computing an integral image of the foreground map, so that a particle can be scored in constant time independently from image resolution.

Furthermore, for each particle, the head cuboid projection on the image plane is classified by a single run of the face-detector. The overlap between the projected head box and all detected faces is used to refine the particle's score. In the same manner, upper body detection is incorporated to support the track.

Using this tracking scheme, computationally expensive features are evaluated locally at the particles' projected positions in the respective images. Thus, the complexity of the tracking algorithm is related linearly to the image size, the number of cameras, and the number of particles. The average tracking error was evaluated to be about $23cm$ throughout all video sequences.

3.2 Head Alignment

Since the estimated position of the lecturer given by the tracking module does not provide consistently aligned bounding boxes of the lecturer's face, a further face alignment step becomes necessary, before faces can be extracted for later processing.

In order to align and extract the lecturer's face in each camera view, we use a frontal and profile face detector which are based on Haar-feature cascades as proposed in [10]. The search space for these face detectors is limited to a search window around the initially estimated position of the lecturer, projected into the respective camera view (see Figure 4 for an example - the big boxes around the lecturer's head depict the search windows for the face detectors).

Since the face detectors sometimes fail to detect a face, we predict the face bounding boxes in those camera views, in which the lecturer's face could not be detected in order to get a facial view for later pose estimation. This can be done if the face was detected in at least two other camera views. From the detected faces, we then compute the lecturer's 3D position by triangulation and project a 3D cuboid around the 3D head location into those camera views where no face was detected (see also Figure 2).

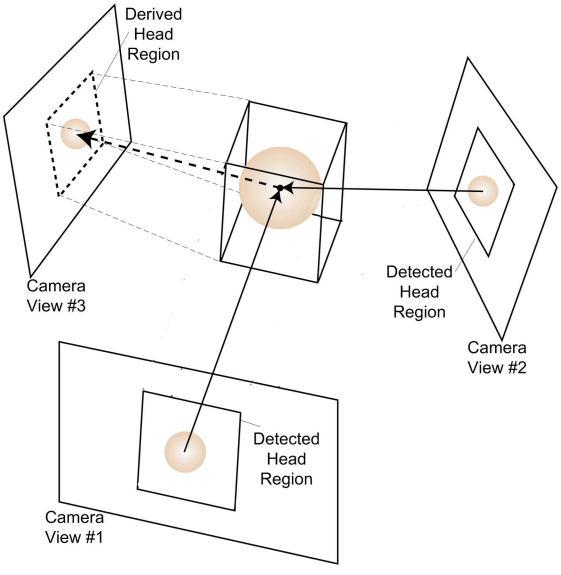

Fig. 2. We detect heads using both frontal and profile face cascades in each camera view. The 3D position of the head is computed by triangulating the centroid of each detected head region in associate camera pairs and searching for the 3D position with the smallest residual. In camera views where no head was found at all, the head's region is derived by placing a fixed-size cuboid around the computed head's centroid in 3D and re-projecting the cube onto the corresponding camera's image plane. The edges of the projection describe the derived head region then.

If - due to false face detections - more than one face is detected in a camera view, those face bounding boxes that lead to the minimal triangulation residual are chosen as the correct ones.

3.3 Classifying Frontal Views vs. Views at the Head's Back

In our experiments, we observed that neural networks for head pose estimation performed worse if views of the back of a head (showing hair only) were included in the training data set. Therefore, we try to automatically detect back-views of heads in our data. To do this, we trained a neural network classifier which outputs the a-posteriori probability, that a given image depicts a frontal view in the range from left to right profile ($[-90°, +90°]$). Following the work we presented in [11], we use a three-layered, feed-forward network, trained with frontal views and views of the head's back only. For the latter the target output was defined to be 0, else 1. Finally, we use a likelihood threshold of 0.5 above which all captures are classified as (near-) frontal views of the head. As input to the neural net, a histogram-normalized grayscale image of the head as well as horizontal and vertical edge images were used. All these were downsampled to 16×16 pixels each, and concatenated into one single feature vector.

The network was trained using standard error backpropagation, minimizing the output error on a cross evaluation set within 100 training cycles.

3.4 Single-View Head Pose Estimation

We first try to estimate the lecturer's head orientation relative to each camera position in the range of $[-90°, +90°]$. Doing the estimation relative to each camera (position) first - instead of estimating head orientations relative to the world coordinate system - allows us to train and use only one single neural network to estimate head pose for all cameras (see Figure 3). This has the advantage that all available facial images from all cameras can be used for training the network. Also, this makes our system independent of the positioning of the cameras in the room and allows us to add further cameras without the necessity of retraining the network.

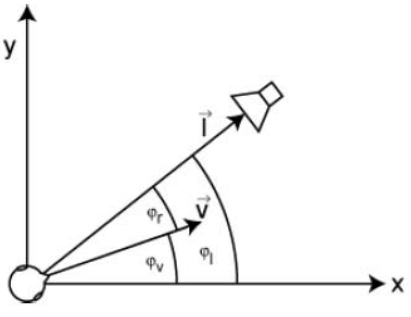

Fig. 3. Each camera's line of sight points straight to the corresponding head's 3D centroid. By using relative head pose angles, the very same head pose estimating neural network may be used for each additional camera view, thus preventing the necessity to train one single network for each camera view respectively.

To estimate head pose, we follow our previous work [11] using a three-layered, feed-forward network with one single output unit. Head pose is estimated continously in the range of $[-90°, +90°]$. As input images, again downsampled histogram normalized grayscale images as well as horizontal and vertical edge images of heads are used.

The network is trained with standard error backpropagation, using a dataset that consists of frontal views of the head only, ranging from left to right profile. As noted above, we experience a more robust performance of the system by limiting its output and therefore the training data to the $[-90°, +90°]$ range.

3.5 Building the Joint Hypothesis

We define $\Theta = \{\theta_i\}$, with $\theta_i \in \{0°, 45°, \ldots, 315°\}$ as the set of all possible head pose classes. These are defined to be relative to the world coordinate system in the room.

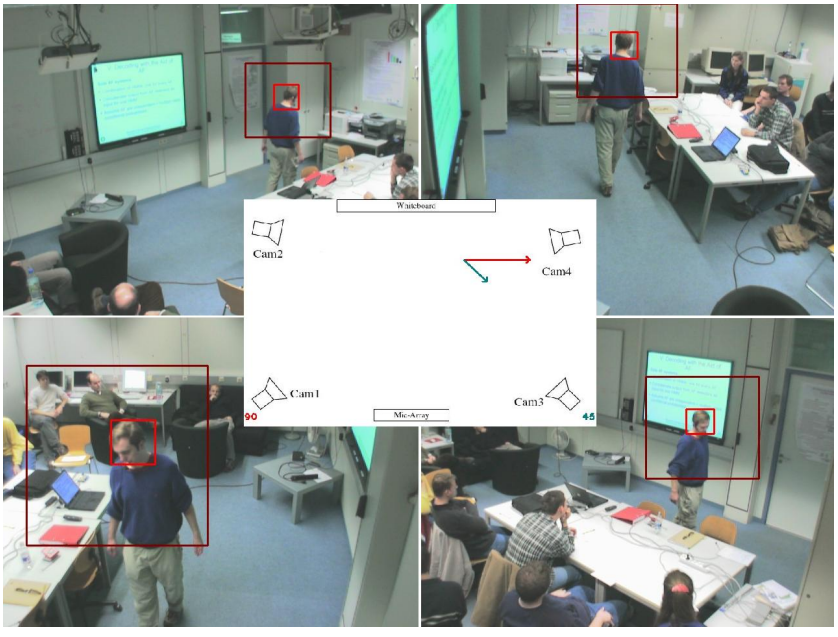

Fig. 4. Example output of our discrete head pose estimation system. The arrows indicate the final head pose estimation (long red arrow) and the groundtruth head pose (short green arrow). Further, the position of the arrows indicate the position of the user in the smart-room. The rectangles in each camera view indicate both our search window (large rectangle) in which the system tries to detect a head, and the actual detected head region (small rectangle).

Further, at each timestamp we have $H = \{h_1, h_2, \ldots, h_n\}$, the set of all single orientation estimations made. n represents the number of cameras used, depicting frontal views at the lecturer's head only.

In making a final decision about the true head pose, we score a pose hypothesis θ_i by summing up the a-posteriori probabilities of all available estimations as follows:

$$\pi(\theta_i) = \sum_{j=1}^{n} P(\theta_i|h_j) \tag{1}$$

Finding the best hypothesis then consists in maximizing the score by searching for the best fitting hypothesis $\hat{\theta}$:

$$\hat{\theta} = \arg\max_{\theta_i \in \Theta} \pi(\theta_i) \tag{2}$$

The described procedure guarantees increasing hypothesis scores, the more camera views are being used and easily allows to extend an existing setup by adding more cameras in order to stabilise the estimation. Hereby, the algorithm's complexity increases linearly with the number of cameras C and the number of possible head orientation classes according to $O(C \cdot |\Theta|)$.

The a-posteriori probabilities in equation (1) are derived from confusion matrices that were built for each camera whilst evaluating the classification performance of the trained neural network on the cross evaluation set. Since confusion matrices transcribe the amount of estimated facial views when the true head pose is known, they allow to compute the a-posteriori probabilities of pose classes when a specific single estimation is given. That way, the posterior probability of a class θ_i given the observation h_j can be computed as

$$P(\theta_i|h_j) = \frac{k_{ij}}{\sum_m k_{mj}} \tag{3}$$

where k_{ij} denotes the matrix element in row i and column j. While the matrix columns define the different estimation classes and the rows describe the groundtruth head pose classes.

4 Experiments and Results

Considering the educational smart-room scenario we already described earlier, we evaluated our implementation on real videos that were recorded during a seminar lecture in 2003. Overall we recorded 7 persons, further splitting each recording into 4 segments of 5 minutes each, on which training and evaluation was realised separately. However, in order to reduce redundancy, we annotated and evaluated every 10th frame only. In the multiuser scenario, we trained the underlying neural networks on segments 1 and 2, using segment 3 as cross evaluation set. Segment 4 was used for evaluation purposes, thus evaluating the networks with video data that has not been seen before in the training stage, though resulting from the same persons. In the unknown user scenario, we implemented a round robin evaluation, thus excluding a person's recording from training and cross evaluation when evaluation is being done on this person's video data.

For providing groundtruth information regarding the true head pose, we manually annotated the videos with the observed head pose of the lecturer, classifying the head's pose manually into one of eight equidistant classes such as $0°, 45°, 90°, ...315°$.

4.1 Multiuser System

In case of the multiuser system, the networks have been evaluated with the same persons they have been trained with, although not the very same segments have been used. In this case, with the use of our earlier described head position tracking module, classifying frontal views of the head performed with an accuracy of 83.5%.

As Table 1 shows, head orientation estimation performed correctly with approximately 59% in our fully automatic scenario. This means, the networks were evaluated using unsupervised head extractions, thus including outliers and variance resulting from imperfect alignment of the corresponding bounding box.

In case of manually annotated 3D positions of the head's centroid and manual removal of extreme outliers, the performance increased to approximately 74% correct detection of the pose class thus showing the impact of imperfect face detections and outliers in the complete system.

One major problem regarding outliers in alignment comes from views where the head region has been derived from other views: Using a fixed-size cuboid surely is ineffective in assigning a hard edged region of interest. Estimating the head's approximate size in 3D and using a secondary triangulation step regarding the bounding box' vertices might therefore provide an alternative for future work.

Table 1. System's performance on head pose estimation in percent. We evaluate both the results for automatic recognizing of facial views as well as choosing near frontal views manually. Using a fully automatic system, the correct pose class is detected 58.9% of the time. In 91.7% of the time, the correct class or a neighbouring pose class is detected ($error < 45°$). When choosing frontal views manually, the correct head pose is recognized in 74.6% of the time.

	correct class	correct or neighbour class
multiuser manual view selection	74.6	96.4
multiuser automatic view selection	58.9	91.7
unknown user automatic view selection	48.4	82.9

The second limitation is clearly resulting from the error produced by the facial view classifier, especially considering the fact how such a false positive classification shifts the range of possible head poses that are to be considered. If, for example, camera 1 and 2 depict frontal captures of the user's head and camera 3 estimates a false positive, the range of possible head poses clearly shifts up to the third camera's view range. Since the orientation estimation network only gets trained up to profile faces captures, the output of the third camera is taken into account as if the head is truly rotated into that direction. This leads to the best matching head pose for a wrongfully extended range, which clearly produces false hypotheses in the end. Regarding these outcasts, temporal filtering could help in reducing this negative effect.

4.2 Unknown Users

The unknown user scenario was realised by implementing the leave-one-out method, where one person was removed from the training data set and exclusively used for evaluation purposes only. The results are shown in Table 1. In this unknown user scenario, the initial facial view recognition step achieved a correct recognition rate of 79.6%.

Overall, correct pose class detection was achieved in 48.4% of the time. In 82.9% of the time the estimated pose class fell in either the correct class or a neighbouring class ($error < 45°$). Although the obtained performance is worse than in the multiuser case, we can see that - as in the multiuser case - the performance increases as more facial views are available for pose classification (see Table 2). The results furthermore indicate that it might be advantageous to train the system with much more data in order to increase the networks' capability of generalisation on unseen people.

Table 2. Correct classification in percent in case of both a multiuser and an unknown user scenario. In both cases, using more frontal views at the head enhances the system's performance.

	1 frontal view	2 frontal views	3 frontal views	avg.
multiuser	37.5%	58.3%	72.5%	58.9%
unknown users	27.3%	55.2%	55.3%	48.4%

5 Conclusions

In this work, we have presented an approach for estimating the horizontal head orientation of a lecturer in a multi-camera smart-room environment. We estimate head orientation in each camera view using a neural network. Multiple head pose estimates coming from various camera views are then fused in order to obtain a more accurate estimate of the lecturer's head orientation.

Since head pose is initially estimated with respect to each camera, our approach is flexible and allows for easy change of camera positions and use of additional cameras without the necessity of retraining the system.

We conducted experiments on a set of real seminar recordings. Our experiments show that the overall error significantly decreases as more facial views are included in the estimation. In a multiuser evaluation, the correct pose class could be detected in 58.9% of the frames. In 91.7% of the time, the correct class or a neighbouring pose class (i.e. a 45 degree error) were estimated. In case of unseen users, in 48.4% of the frames the pose class was correctly determined (82.9% when including the neighbouring pose classes).

Our setup provides an unobtrusive estimation of a lecturer's rough head orientation. We believe that this will be useful for many applications in smart seminar rooms, e.g. in order to detect people's focus of attention and interaction among each other.

As our experiments show, pose estimation results were quite heavily affected by false detections of near frontal facial views. In future work we will try to circumvent these problems by soft classification of near frontal views (instead of hard decisions as it is the case right now). We furthermore hope to improve results by using temporal filtering to stabilize the system output.

Acknowledgements

This work has been funded by the European Commission under contract nr. 506909 within the project CHIL (http://chil.server.de).

References

1. S. O. Ba and J.-M. Obodez. A probabilistic framework for joint head tracking and pose estimation. In *Proceedings of the 17th International Conference on Pattern Recognition*, 2004.
2. A. H. Gee and R. Cipolla. Non-intrusive gaze tracking for human-computer interaction. In *Proceedings of Mechatronics and Machine Vision in Practise*, pages 112–117, 1994.
3. T. Horprasert, Y. Yacoob, and L. S. Davis. Computing 3-d head orientation from a monocular image sequence. In *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition*, 1996.
4. R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *Proceedings of the IEEE International Conference on Image Processing*, 2002.
5. K. Nickel, T. Gehrig, R. Stiefelhagen, and J. McDonough. A joint particle filter for audio-visual speaker tracking. In *International Conference on Multimodal Interfaces ICMI 05, Trento, Italy*, 2005.
6. C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 246–252, 1999.
7. R. Stiefelhagen, J. Yang, and A. Waibel. A modelbased gaze tracking system. In *Proceedings of the IEEE International Joint Symposia on Intelligence and Systems*, pages 304–310, 1996.
8. R. Stiefelhagen, J. Yang, and A. Waibel. Simultaneous tracking of head poses in a panoramic view. In *International Conference on Pattern Recognition*, 2000.
9. Y.-L. Tian, L. Brown, J. Connell, S. Pankanti, A. Hampapur, A. Senior, and R. Bolle. Absolute head pose estimation from overhead wide-angle cameras. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2003.
10. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.
11. M. Voit, K. Nickel, and R. Stiefelhagen. Multi-view head pose estimation using neural networks. In *Second Workshop on Face Processing in Video (FPiV'05), in Proceedings of Second Canadian Conference on Computer and Robot Vision. (CRV'05), 9-11 May 2005, Victoria, BC, Canada*, 2005.