Multiview Social Behavior Analysis in Work Environments

Chih-Wei Chen
Department of Electrical Engineering
Stanford University
Stanford, California 94305
Email: louistw@stanford.edu

Hamid Aghajan
Department of Electrical Engineering
Stanford University
Stanford, California 94305
Email: aghajan@stanford.edu

Abstract—In this paper, we propose an approach that fuses information from a network of visual sensors for the analysis of human social behavior. A discriminative interaction classifier is trained based on the relative head orientation and distance between a pair of people. Specifically, we explore human interaction detection at different levels of feature fusion and decision fusion. While feature fusion mitigates local errors and improves feature accuracy, decision fusion at higher levels significantly reduces the amount of information to be shared among cameras. Experiment results show that our proposed method achieves promising performance on a challenging dataset. By distributing the computation over multiple smart cameras, our approach is not only robust but also scalable.

I. INTRODUCTION

Camera networks have been widely deployed for many applications, including security surveillance and traffic monitoring. Meanwhile, progress in computer vision has enabled the automation of content analysis on the data captured by these sensors. As the size of the network grows and the complexity of the information to be extracted increases, a centralized processing paradigm seems incompatible. Distributed algorithms emerge as the more suitable solution to the problem, where raw data are processed locally. This mitigates the problem of limited network bandwidth and the computing capacity of a single processor.

In this paper, we develop algorithms for social behavior analysis of people in workplace environments. The proposed methods are designed for distributed smart cameras. In particular, we design a tracking algorithm that tracks multiple people in a real, unconstrained environment. Human head poses are then estimated, and a discriminative interaction detector discovers the dyadic social interactions in that environment. The processing is done locally at each camera. Information extracted is then exchanged across the camera network to obtain more accurate, reliable decisions.

The study of proxemics [1] has revealed the close correlations between physical distance and social distance. In that sense, locating people not only assists the understanding of how people utilize a given space, but also brings to light how interactions are conducted. However, tracking multiple people is a challenging task, especially in crowded scenes where people are often occluded and the background is cluttered. We follow a tracking-by-detection paradigm [2, 3] that has been widely used. An efficient Chamfer matching [4] compares an Ω-shape head-shoulder template [5] against the edge map of the incoming video frame. A low-level tracking algorithm bridges the gaps between detections. Results from each camera’s tracker are combined, using the geometric homographies among cameras as constraints. The most probable locations of the head, i.e. the location most cameras agree on, can then be determined. The use of multiple cameras helps resolve the ambiguity caused by occlusion.

To understand human behavior, the focus of attention provides important information and evidence. It reveals not only how people interact with each other, but also how people interact with the ambient environment. The gaze direction is arguably the most important indicator of a person’s attention. Head orientation contributes much to gaze direction, and focus of attention estimation based on head orientation has been shown to have high accuracy [6]. Therefore, we propose to extract the head poses for the understanding of human social behavior, as the head orientation indicates the intended target of interaction.

For surveillance cameras, the faces are usually at a distance and therefore small in the acquired videos. As a result, accurate extraction of facial features is difficult, if not impossible. We adopt an image patch based method, where the head pose is estimated from normalized head images via learning. In this work we consider only the yaw rotation of the head pose. The estimation problem is treated as a classification problem, where the head orientation space is discretized to different classes. Image segmentation divides the image into skin, non-skin foreground, and background pixels. Randomized ferns [7], a special type of randomized decision tree algorithm, are trained to classify the image patch to one of the head poses. We extend the method to the multi-view framework [8], and show the improvement over monocular results.

The area we would like to address is human interaction. Automatic extraction of such information not only facilitates semantic video retrieval but also allows us to answer questions such as “how frequently do they talk to each other?” or “where do the group meetings take place?” The type of interaction we focus on in this paper is direct, primitive interaction between two people. Such interaction has two prominent features:
proximity and eye contact. We train a discriminative classifier on dyadic interactions using the distance and relative head pose as the feature.

The contributions of this paper are as follows. First, we present a distributed framework for human localization, head pose estimation and social interaction detection. Second, for each task we propose algorithms tailored for distributed camera systems where each camera processes the video locally and fusion is done in a higher abstraction. Finally, we compare the performance of our proposed approach to single-view results, and evaluate the performance of the distributed algorithms at different levels of information fusion. The evaluations are carried out on a real, challenging dataset where people interact naturally in a real office environment.

II. RELATED WORK

Extensive research has been devoted to human detection and tracking from a single camera [2, 9]. The most challenging problem in monocular detection and tracking is occlusion. To address this problem, multiple cameras are used. New algorithms have been developed, and existing ones extended, for the multi-view scenario [10, 11]. Researchers attempting to understand human motion have also adopted multi-camera framework, including human pose recovery [12] and activity recognition [13]. Human behavior in an office environment has been studied by using prior knowledge about the layout of the environment to associate places to activities [14]. Instead of comprehensive understanding of human activities, we focus on the interactive behavior of people. Social interactions can be identified by analyzing people’s walking pattern [15]. In [16], social groups and their leaders are identified by linking people using their similarity in motion. We address a more static work environment where social links are determined by people’s location and focus of attention.

Existing research on head pose estimation mainly falls into one of the following categories. Template-based [17] methods match the incoming image to a set of templates, and estimations are assigned by the most similar ones. Learning-based methods [18, 19] learn the function between the input image and the pose, either via classifiers or regressors. Embedding-based methods [20, 21] discover the low dimensional manifold and the embedding. A comprehensive survey can be found in [22]. Because of the nature of the applications, e.g. human computer interaction (HCI) and driver visual attention monitoring [23], most research focuses on head pose in the range of $-90^\circ$ to $90^\circ$. Our pose estimation considers full pose range, where people move around freely in the test environment. We learn randomized ferns using cues from skin color and background segmentation, and extend it to the framework of multiple cameras.

III. SOCIAL BEHAVIOR ANALYSIS

In this section we describe our approach to extract the necessary information for social behavior analysis. In particular, we detail the tracking algorithm, review the random ferns classifiers and their application to head pose estimation, and finally discuss how location and head orientation are used to detect human interaction.

A. Tracking

We first explain how tracking is done on each individual camera. Instead of full body tracking, only the head is tracked because of its direct applicability for head pose estimation. Moreover, the body is often occluded in a work environment, e.g. behind the desk or chair. To track multiple people in the environments, we adopt a tracking-by-detection approach as in [24]. The tracker integrated two methods complementary to each other: detection association and low-level tracking. For detection, foreground mask is first obtained by constructing an adaptive Gaussian-mixture background model for the incoming video stream. An $\Omega$-shape head and shoulder silhouette template is then matched against the edges from foreground objects. The template matching is performed at multiple scales using Chamfer distance, and thus is robust to background clutters and appearance variations.

Detection is associated with the current track based on both its geometric overlap with the current bounding box, and its similarity with the current appearance template. The similarity is measured by the normalized correlation of the two image patches. This appearance attribute is also the feature our low-level tracking is based on. When there is no detection, we search in a local region near the current track location, and find the image patch that achieves maximum correlation with the current track. If the maximum score is below a threshold, the previous image patch is kept as the appearance template. Otherwise the appearance template is updated by the image patch at the new location. The two modes are complementary: the low-level mode fills the gap between detections (due to occlusion or static foreground merging into the background model), and the detection mode recovers the track once the target reappears.

While biometric recognition methods have enabled the identification of people from visual sensors, we incorporate an RFID system to help the initialization of the tracker. An RFID sensor and an auxiliary camera are installed at the entrance of the office. When a person enters the office, the RFID he carries triggers the recording system, and a color histogram is built to model the appearance of that person on that day. The person is then tracked throughout the day. Our setup also enabled the dataset collection of people’s appearance over time, which can be subsequently used to learn a recognition system.

B. Localization

Once the head location of the target person in the image is known, his location can be estimated by mapping the 2D image location to real world 3D coordinates. In monocular view, with a calibrated camera, this can be done if the height of the target is known. In an office environment, people are either standing or seated most of the time. Other poses only appear briefly and do not contribute much to the overall statistics of the behavior analysis. Therefore, two heights are used to estimate the location, one for standing and one for seated people. Most
of the time only one of the two hypotheses would be valid. In the case of ambiguity, the location closer to the previous estimated position is used.

When multiple views are available, we use the following method to combine information from all views. Let $I_0, I_1, I_{n-1}$ be the images obtained from $n$ cameras. Without loss of generality, let $I_0$ be the reference image. Denote the observation at location $l$ in the reference image $x$. Its corresponding observations in other views, $x_1, \ldots, x_{n-1}$ can be found by computing the homography to the respective images. Let $X$ be the event that the pixel at location $l$ lies in the target’s head $H$. Following the formulation in [10], the probability of such an event given the observations $x_0, x_1, \ldots, x_{n-1}$ can be written as

$$P(X|x_0, x_1, \ldots, x_{n-1}) \propto P(x_0, x_1, \ldots, x_{n-1}|X)P(X)$$

$$\propto P(x_0|X)P(x_1|X) \cdots P(x_{n-1}|X)$$

where Bayes’ law is applied in the first line, and the second line follows assuming conditional independence among different cameras. By the homography constraint in [10], the probability of $P(x_i|X)$ is proportional to $f(x_i)$, the probability that $x_i$ belongs to the foreground head. In [10] the planar homography between cameras is derived from ground plane to identify the feet. In our case, the homography is calculated from the plane parallel to the ground at the target’s height, since the region of interest is the person’s head.

In our framework, a probability is assigned to the head location based on the mode the tracker operates in. In detection association mode, the detection score is converted to a probability. During low-level tracking, the probability is proportional to the correlation score. The probability indicates the likelihood that a region belongs to the foreground head of the target person, and is combined across cameras using the method above. More specifically, a synergy map is constructed by choosing a reference frame, warping the likelihood map of other views to this reference view, and multiplying the likelihood together. An example is shown in Figure 1.

C. Head Pose Estimation

Here we treat the head pose estimation as a classification problem, where the 360 degrees yaw rotation space is uniformly discretized into $N$ distinct views. We show in Figure 2 the 8 classes classification, and example head images for each class. Head pose estimation is sensitive to image alignment. To ameliorate the problem, an ellipse is fitted to the contour of the head, and the image patch centered at the ellipse is used as the input to the classifier. The scale of the ellipse is varied depending on its distance to the camera.

Randomized ferns [7] have been applied to the estimation of head poses [25]. A fern is a special type of decision tree where the tests at equal depth are the same. Assume $F_1, F_2, \ldots, F_n$ are the ensemble of randomized ferns, $I$ the head image to be classified, and $C = \{c_1, \ldots, c_N\}$ the class label. Each leaf node in the fern carries a posterior probability learned from the training examples. The image $I$ traverses down ferns according to the decisions at each interior node. Suppose the image reaches a leaf node in the $k$th fern with the posterior $P(C|F_k, I)$. The head patch is then classified to class $h$ such that,

$$h = \arg \max_j P(c_j|F_1, \ldots, F_n, I)$$

$$= \arg \max_j \sum_{i=1}^n P(c_j|F_i, I).$$

Arithmetic mean is taken to combine estimations from all ferns because of its proven effectiveness and robustness to bias [25].

We now describe how the ferns are constructed and how the posteriors are learned. The training examples are cropped head images normalized to a predefined size, and manually labeled the ground truth orientation. Each pixel in the image patches is segmented into one of the three labels: skin, non-skin foreground, and background. Background pixels are obtained from background segmentation, and skin pixels are identified by a color-based skin detector in HSV color space. The test at
each fern node is then based on the label. For example, the test could be whether or not the pixel at location \((x, y)\) is a skin pixel. The locations and pixel labels are randomly selected to construct the random ferns. Examples of the segmented head images and randomized ferns can be found in Figure 3.

The training examples are passed down the \(k\)th fern to reach the leaf nodes. Each leaf node keeps a histogram of the class label. After all training examples are processed by the fern, the posterior \(P(C|F_k, I)\) for a given image \(I\) is computed by normalizing the histogram at the node \(I\) reaches. The class label can then be estimated from the equations above.

![Figure 2: Head images are classified to one of the eight orientation classes. Examples of head images tracked by our system are shown. The resolution of the head image is low, and the appearance of the target people exhibits a large variation.](image)

IV. DISTRIBUTED SMART CAMERAS APPROACHES

In this section we describe how our algorithms are applied, or extended, to the distributed smart cameras framework. Location, head pose, and interaction estimated by each smart camera are combined to reach a more accurate decision. Information from distinct views can be fused at various levels, each requiring different amounts of data to be transmitted over the network. We explore several possibilities and their corresponding algorithms.

A. Centralized Processing

In a traditional setup, all cameras stream the raw data to a central processor. Therefore, the scalability of the sensor network is limited to the bandwidth of the underlying network and the processing capacity of the central processor.

B. Location and Head Pose Fusion

The localization method we employ does not require images from different views be aggregated at a central processor. In fact, only the homography and the tracked head location from the cameras that share an overlapping field of view are needed for a local camera to leverage synergies among individual sensors and obtain a more robust estimation. Since only static cameras are considered, the field of view overlap is determined by the geometric configuration of the cameras and does not change with the dynamic scene. When a target is not detected in a view, an uniform distribution will be returned if an estimation is requested by other cameras. Therefore the decision will not be affected by missed detections.

The randomized fern can be readily extended to a multi-view scenario. Using the same notation from the previous section, a head patch is classified to the angle class \(h\) that satisfies

\[
h = \arg \max_{j} \sum_{k \in \mathcal{N}} \sum_{i=1}^{n} P(\phi_k(c_j)|F_i, I).
\]  (1)

where \(\mathcal{N}\) is the set of cameras where the target is observable, including the local camera itself. The function \(\phi_k(c_j)\) maps the orientation class in the local camera to the correct representation in the \(k\)th camera, relative to its line of view. The distributions estimated by each camera are averaged together. An angle class is given a higher posterior probability if more cameras agree on it.

Equation 1 is equivalent to having multiple observations, and classifying them using sets of randomized ferns. Without actually sending the image patches, the proposed method achieves the same performance as a centralized approach while distributing the computation across different cameras.
C. Interaction Fusion

Finally, we consider information fusion at the interaction decision level. That is, each smart camera detects interactions, and only detection results are combined. This requires the least amount of data to be shared for the cameras to collaborate in interaction detection. The intuition is that false positives by a single camera can be recovered if other cameras have correct classification results.

A common approach to combine estimation from multiple cameras is to take the average [8]. However, we argue that not all estimations should have equal importance in the final decision. For example, if from one view the camera has detected that the target is occluded, the pose estimation, and hence the interaction detection, are likely to be erroneous. We therefore propose a probabilistic method where each camera reports a confidence score for itself, as well as for its neighboring cameras based on the knowledge of their relative geometric configuration. The final decision is made by taking into account all the estimations and their respective confidence scores.

Assume that for an example, the output score from the $i$th view’s SVM classifier is $h_i$. Its confidence score for itself is the same probability used for constructing the synergy map during joint localization. This score is given by the tracker and reflects its confidence in localization. Denote this score $w_{ij}$. Notice that the dyadic interaction classifier operates on a pair of people, so the average of the two localization probabilities is taken as the score. Based on its own estimation of the head poses and their relative location, the $j$th camera makes a decision on the interaction and has an SVM score of $h_j$. The weight on the $j$th camera assessed by the $i$th view, $w_{ij}$, is proportional to the classification accuracy of the head pose estimator for a particular pose. The weight is learned during training of the head pose estimator, and has a multimodal distribution for different local head pose estimation. The idea is that some head poses are more difficult than others, e.g. the back of the head is less discernible than frontal face. Again the weight is the mean of two weights, for each of the person in the interaction pair to be evaluated. The final score combining all $n$ views is then

$$H = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} h_j. \quad (2)$$

V. Experimental Results

As opposed to experiments conducted in controlled environments with people instructed to act in a certain manner, we evaluate our methods on a challenging dataset [24] of videos captured in a real office where people behave naturally. Three AXIS network cameras are installed in 3 different corners in the office. The layout of the office, as well as sample frames from the cameras, are shown in Figure 5. The cameras capture VGA ($640 \times 480$ pixels) video at 30 fps, and are synchronized to a Network Time Protocol (NTP) server.

Test data. The system captures the daily life of people working in the office environment, where people greet, talk and have meetings occasionally. Tens of hours of video have been recorded. A short sequence is extracted from the recordings for testing. The test video sequence is about 11 minutes and 30 seconds long, and contains 5 people interacting in diverse ways. The 5 subjects have very different appearance attributes, such as hair style and facial features, making head pose estimation a challenging task. The locations, head poses and primitive interactions in the test video are manually labeled for evaluation.

Localization. We quantify the localization accuracy. A track is considered correct if the overlap of the ground truth and the tracked bounding box is at least 50% of both rectangles. There are 5 tracks for the 5 people in the video. Ground truth head bounding boxed are annotated for all 5 tracks in all 3 views for the entire video, sampled at 6 fps. Each view contains 4150 synchronized frames.

We use Cam 0 as the reference frame. The synergy map is warped to it, and we report the performance of the multi-view method using the ground truth label on the reference frame. Using information from multi-view cameras, the precision improves from 93.8% to 97.8%. We show the number of false positives for each view in Figure 4. The reduction in false positives reflects the multi-view system’s ability of resolve occlusion.

![Localization results](image)

Fig. 4: Localization results. We plot the false positives of all five tracks. We compare the performance of each individual monocular camera to the results obtained from using all 3 views. Using information from multi-view system greatly reduces the number of false positives.

Head pose estimation. We compare the head pose estimation performance from single-view and multi-view. We compare our method to several baseline methods. First of all, the head pose can be estimated from monocular view. That is, each camera classifies the head patch it discovers. The overall accuracy for all 3 cameras is 0.53. A closer look at the failure cases suggests that incorrect localization by the tracker contributes to many of the misclassified cases. We therefore consider a second baseline, where each camera first agrees on the best location of the target. The synergy map is then projected back to each view, and an ellipse is fitted to the new location. The classifier then uses the new location as the input. The accuracy is improved to 0.61. While a better localization mitigates the problem of misalignment, it still cannot recover the correct orientation if the target person is occluded in a local
view. Multi-view system must be used to obtain a more reliable estimation. Our multi-view head pose estimation achieves an accuracy of 0.69.

Interaction. The test sequence contains several instances of different forms of interactions. At the beginning of the sequence, two people talked to each other in their office chairs. Then the other three started a meeting among themselves, and people turned their faces when engaging in conversations with different people. After both meetings were dismissed, two people began working together at the same desk. Finally, a group discussion involving all people took place at the center of the office. We sample dyadic interaction instances from the sequence. About 200 frames are used as the test set, and we run our interaction classifier on it. Examples frames and discovered interactions are shown in Figure 5.

The average precision from a single view camera, Cam 0, is 0.71. The multi-view system that uses the aggregated location and head pose estimation from all cameras as the input to the trained SVM achieves an average precision of 0.83. If only interaction decisions, made independently by each camera based on its own estimation of location and head poses, are combined, the average precision is 0.79. Information fusion at interaction decision level has comparable performance with the full multi-view method, and only requires little information be shared among the cameras in the network.

VI. CONCLUSION

We introduce a multi-view framework for the understanding of human social behavior. The performance of the proposed methods has been demonstrated on a challenging dataset. In particular, we have seen that information fusion at a highly abstract level can achieve promising results, making distributed smart cameras a compelling framework for its robustness and scalability.

Currently interactions are detected independently on each frame. Temporal information can be incorporated to remove noise, and to provide more insight about the interactions. As future work, we plan to further analyze the structure of the interactions, e.g. group meetings, presentations, and the social structure, such as each individual’s social role, based on our research results.

REFERENCES


