# JOINT FACE AND HEAD TRACKING INSIDE MULTI-CAMERA SMART ROOMS

*Zhenqiu Zhang*[†,1], *Gerasimos Potamianos,*[2] *Andrew W. Senior,*[2] *Thomas S. Huang*[1]

[1] Beckman Institute, University of Illinois, Urbana, IL 61801, USA
[2] IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
Emails: zzhang6@uiuc.edu, {gpotam,aws}@us.ibm.com, huang@ifp.uiuc.edu

## ABSTRACT

The paper introduces a novel detection and tracking system that provides both frame-view and world-coordinate human location information, based on video from multiple synchronized and calibrated cameras with overlapping fields of view. The system is developed and evaluated for the specific scenario of a seminar lecturer presenting in front of an audience inside a "smart room", its aim being to track the lecturer's head centroid in the three-dimensional (3D) space and also yield two-dimensional (2D) face information in the available camera views. The proposed approach is primarily based on a statistical appearance model of human faces by means of well-known AdaBoost-like face detectors, extended to address the head pose variation observed in the smart room scenario of interest. The appearance module is complemented by two novel components and assisted by a simple tracking drift detection mechanism. The first component of interest is the initialization module, which employs a spatio-temporal dynamic programming approach with appropriate penalty functions to obtain optimal 3D location hypotheses. The second is an adaptive subspace learning based 2D tracking scheme with a novel forgetting mechanism, introduced as a means to reduce tracking drift and increase robustness to illumination and head pose variation. System performance is benchmarked on an extensive database of realistic human interaction in the lecture smart room scenario, collected as part of the European integrated project "CHIL". The system consistently achieves excellent tracking precision, with a 3D mean tracking error of less than 16 cm, and is demonstrated to outperform four alternative tracking schemes. Furthermore, the proposed system performs relatively well in detecting frontal and near-frontal faces in the available frame views.

***Index Terms***— Person tracking, face detection, face tracking, multi-camera tracking, dynamic programming, adaptive subspace tracking, mean-shift tracking, AdaBoost, background subtraction, triangulation, lecture data, smart rooms.

## 1. INTRODUCTION

Visual detection and tracking of humans is an important problem with numerous applications that range from automated surveillance to interfaces for human-computer interaction. In general, robust human tracking in complex scenes is challenging. In some circumstances however, multiple time-synchronous and calibrated camera sensors with overlapping fields of view may be available, from which both frame-view and world-coordinate human location information can be derived. In such scenarios, efficiently combining frame-level appearance-based human detection with temporal and spatial constraints constitutes a viable approach that can simultaneously provide both desired types of location information with improved accuracy, while avoiding reliance on any form of background modeling or motion estimation. This paper introduces a novel human tracking vision system employing these principles, developed and evaluated for the specific scenario of tracking a seminar lecturer presenting inside a "smart room" in front of an audience.

This scenario is of central focus in the European integrated project "CHIL" ("Computers in the Human Interaction Loop" [1]). In CHIL, smart rooms have been set up, equipped with multiple audio and visual sensors that include a minimum of four calibrated and time-synchronous cameras with highly overlapping fields of view, located at the room corners. Numerous seminars have been recorded in such rooms providing a large multi-sensory and multimodal database of real human interaction [2]. The resulting CHIL corpus, annotated with a wealth of multimodal information, has been crucial to the development and evaluation of a multitude of technologies for perception of humans in the lecture scenario of interest [3,4]. Prominent among such technologies is the task of locating the lecturer's head position, both in the three-dimensional (3D) space — in the form of head centroid world coordinates, as well as in the available two-dimensional (2D) frame views — as bounding boxes of visible faces [5]. Such location information can be further utilized in support of numerous audio-visual perception technologies: For example, 2D face information is useful for person identification [6], whereas 3D location coordinates can be employed in acoustic beamforming for far-field automatic speech recognition [7], as well as to obtain close-up presenter views based on steerable pan-tilt-zoom cameras [8, 9] or camera selection schemes [10]. The views can further assist identification [11] and audio-visual speech technologies [12], among others, with obvious utility in lecture indexing and understanding of the interaction.

It becomes clear that for the CHIL lecture scenario described above a visual system that combines face detection, tracking, and multi-camera processing is both feasible and desirable. This paper introduces such a system, developed to provide both 2D-face and 3D-head location information of a single person (the lecturer) in CHIL seminars. Like most 3D approaches, the proposed algorithm consists of a sequence of 3D (re-)initialization and tracking phases, with a tracking drift detection mechanism controlling the switch between the two. Similarly to other works, all its stages depend on 2D information from separate views to obtain 3D location world coordinates based on camera calibration [13].

However, the proposed system deviates from other research efforts that focus on the 2D or 3D tracking problems alone, in that it jointly considers them within a single framework, in order to improve both 2D-face and 3D-head localization accuracy. This is accomplished by relying heavily on the appearance model of the

[†] This work was performed while Zhenqiu Zhang was on a summer internship with the Human Language Technology Department at the IBM T.J. Watson Research Center.

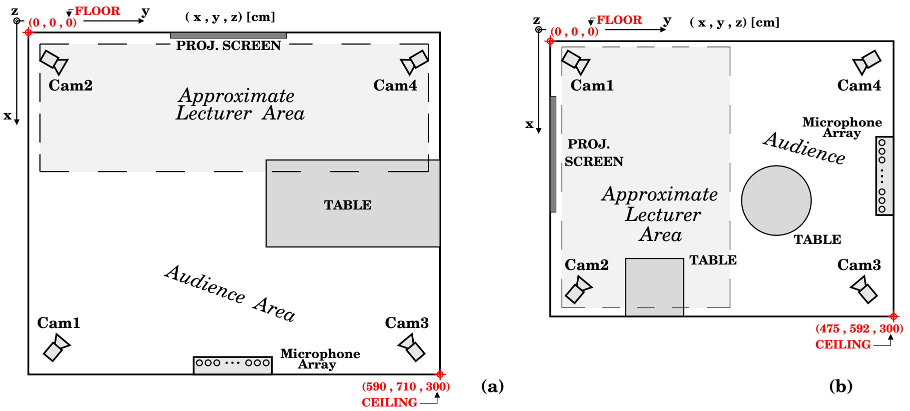

**Fig. 1**. *Overview of the CHIL lecturer video tracking task. Schematic diagrams of the smart rooms located at two CHIL project partners: (a) Universität Karlsruhe (UKA), Germany, and (b) Istituto Trentino di Cultura (ITC), Italy. The CHIL lecture corpus used in our experiments for single-person (lecturer) tracking, has been collected at these two sites.*

tracked object – here the lecturer's head, as viewed in 2D by the available cameras. For this purpose, off-the-shelf statistical classifiers of human faces are utilized, in particular AdaBoost-like face detectors, appropriately extended to address the head pose variation observed in the smart room scenario. As a result, in the developed system, 2D face detection plays a pivotal role in 3D head tracking, being employed in system initialization and in detecting possible tracking drift. Similarly, 3D tracking determines the 2D frame regions where a face detector may be subsequently applied. An additional differentiator of the proposed system is that no form of motion estimation or background modeling is needed. The algorithm therefore remains robust to the unpredictability of motion, occlusion, and background changes in the heavily cluttered CHIL smart rooms. This is in contrast to all alternative tracking systems in the literature (to our knowledge) that address the smart room scenario of interest [14–23].

Two additional components in the proposed system complement the appearance module, implementing a number of novel ideas: One is the initialization module that employs a spatio-temporal dynamic programming approach to obtain optimal 3D location hypotheses. For this purpose, while scoring candidate hypotheses, the adopted implementation penalizes not only large trajectory discontinuities over time, but also accounts for hypothesis appearance similarity between camera views. The second component of interest is a 2D tracking module, used as part of the 3D tracking phase. This component utilizes an adaptive subspace learning based scheme [24]. A novel forgetting mechanism is introduced into this technique, as a means to reduce tracking drift and increase robustness to illumination and head pose variation. Furthermore, this tracking is applied on only two of the four available camera views, selected based on the initialization component. This of course results in significant algorithmic speed-up during 3D tracking.

Finally, the extensive benchmarking of the proposed approach constitutes an important aspect of the paper, breaking away from the toy-problem or small-scale evaluation paradigm that often accompanies other works in the area. In particular, the developed system is benchmarked on all three parts of the CHIL lecture corpus. This is a large database that exhibits significant data variability, with no artificially imposed constraints in the human interaction and behavior patterns, thus allowing meaningful technology development, evaluation, and algorithmic comparisons [2, 5]. Furthermore, the proposed system is compared to a number of 3D tracking methods, ranging from small algorithmic variations of it to significantly different approaches that contain motion estimation or background subtraction components [23, 25].

The rest of the paper is organized as follows: Section 2 briefly discusses literature work relevant to this paper. Section 3 presents a more detailed overview of the tracking task and introduces the proposed system. An in-depth presentation of its components follows in Section 4. Section 5 describes alternative systems considered in our experiments on CHIL lecture data, which are subsequently presented in Section 6. Finally, a brief summary and discussion in Section 7 conclude the paper.

## 2. RELATED WORK

Much work has been devoted to the core problems of human detection and tracking that constitute the focus of this paper. For this purpose, human body models are often used, ranging from simplistic blob appearance or cylindrical shape models [26] to more complex articulated ones [27–29]. An alternative approach to these problems is detecting and tracking human faces.

For face detection, machine learning based techniques are widely considered as the most effective, for example based on neural networks [30], support vector machines [31], network of linear units [32], or the AdaBoost approach [33] that has received much attention in recent years. Alternative methods using traditional image processing algorithms based on color and edge information [34], or optimization to match learned shape and/or appearance to data [35] have also been shown to achieve good performance. Many such techniques can be further extended to handle detecting faces under varying head pose, as for example in [36, 37], where pose-based appearance frameworks are proposed, or the multi-pose face detection work of Li et al. [38], where "FloatBoost", an AdaBoost variant, is employed. The latter approach is used in our proposed system.

Similarly, for tracking faces, various target representations have been used in the literature, such as parameterized shapes [39], color distributions [40], image templates [41] and the eigenspace approach [42], to name a few. Tracking with fixed representations however is not reliable over long durations, and a successful tracker needs to allow appropriate model adaptation. Not surprisingly, a number of tracking methods have been developed to allow such adaptation online, for example the EM-algorithm based technique of [43], the feature selection mechanism of [44], and the paramet-

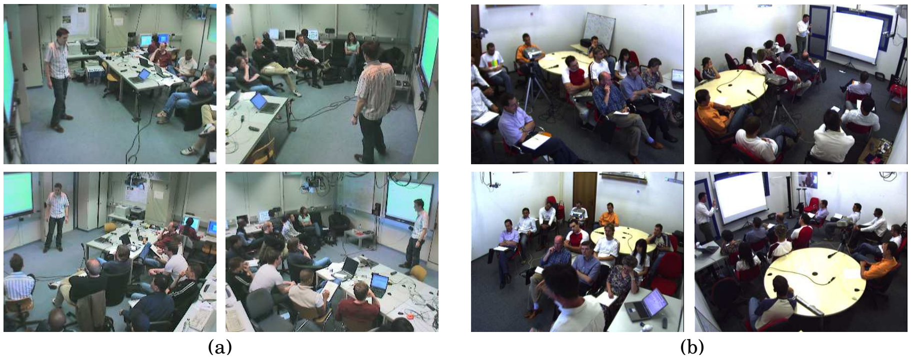

**Fig. 2**. *Examples of synchronous four camera views of the (a) UKA and (b) ITC data, part of the CHIL lecture corpus.*

ric statistical appearance modeling technique in [45]. An interesting non-parametric approach appears in Lim et al. [24], where the appearance subspace is learned online by an efficient sequential algorithm for principal component analysis (PCA), updated with the incoming data vectors. An extension of this technique is employed in our proposed system.

In general however, real interaction scenarios, such as in the CHIL domain, present significant challenges to most face detection and tracking algorithms, for example partially occluded and low-resolution faces, as well as lighting and head-pose variations. These difficulties can often be successfully addressed, only if additional information is available in the form of multi-camera input, in order to reduce spatial uncertainty in the scene [46]. Naturally, some researchers have begun to exploit multiple camera views where they are available, and several tracking systems attempt to fuse information from the available sensors to yield 3D tracking results [9, 47–53], using for example Kalman filters [54], particle filters [55, 56], or just scene and camera geometry [46].

The above ideas have already found their way into a number of papers that address the lecturer tracking problem in the CHIL scenario. In 2D face tracking work reported in [14, 15], statistical face detection is assisted by either a motion model or a combination of foreground-background segmentation [57] and 2D Kalman filtering. However, neither system utilizes 3D information. A few other works aim to provide 3D head information in the CHIL scenario of interest [16–23]. These differ from our proposed system in various aspects, most importantly that they do not focus directly on the 2D face appearance information (with the exception of [23]), but rather model and track larger parts of the human body. For this purpose, they all use background modeling [16–20] or motion information [21–23]. The extracted camera view information is then combined across views by employing either triangulation-based, decision fusion mechanisms [16, 20, 22, 23], or likelihood fusion by means of particle filters [17,19,21]. An alternative technique appears in [58], where histogram features are directly combined across camera views within a 3D kernel based tracking framework — a process akin to feature fusion. That system however lacks an initialization component.

## 3. TASK AND SYSTEM OVERVIEW

As already mentioned in the Introduction, the proposed tracking system constitutes a joint face- and head-tracking approach, developed to address the CHIL lecturer tracking task. In the following, we provide a brief overview of the task, as well as a summary of the proposed approach. A detailed presentation of the system components follows in Section 4.

### 3.1. Overview of the Tracking Task

In the CHIL scenario of interest, a standing subject presents a lecture in front of a small (mostly sitting) audience. The interaction occurs inside a smart room (typically of less than 7.5m×6m×3m in size) equipped with a variety of audio and visual sensors. Among them are four synchronized calibrated cameras with relatively wide-angle and overlapping fields of view, located near the room corners by the ceiling. The cameras are set in such a fashion as to ensure that at least two of them capture the lecturer's head at any given time. In practice, in most instants, two cameras capture lecturer head views that are deemed as "visible faces", namely occasions where the nose and at least one eye are visible, with head poses obviously ranging from frontal to profile. Schematics of two such smart rooms with approximate camera locations are depicted in Fig. 1, and example frames from the four corner cameras in these two rooms are shown in Fig. 2. Notice the relatively constrained region where the lecturer moves, typically close to the whiteboard or projection screen, and limited by furniture and the sitting audience members.

The proposed system jointly addresses two tasks in the above scenario: The first is automatic lecturer localization in the 3D space. Of particular interest is clearly head centroid localization, as discussed in the Introduction. The second task concerns the lecturer face detection and tracking in the four available 2D camera views. The task has been defined in the CHIL project as estimating the bounding box of the "visible faces" of the lecturer in the four camera views, aiming to provide far-field visual recognition of the lecturer's identity.

Both head and face tracking tasks are extremely difficult in this scenario, due to the unconstrained, unstaged data. The smart rooms happen to also be computer labs in operation, with people frequently entering or moving around during the seminars. Occlusions are therefore common. In addition, lighting variations affect the lecturer's visual appearance, due to movement particularly into the projector's beam. Furthermore, the camera resolution is relatively small (typically 640×480 pixels or slightly larger), hence insufficient to cover far-lying smart room space in high resolution; in particular, faces often occupy less than 15×15 pixels in the camera frame views. Finally, the lecturer's varying spatial head orientation and position obviously complicate 2D detection and tracking.

To assist in these tasks, the CHIL lecture corpus provides a significant amount of development data, collected in the smart rooms and scenario of interest, that has been manually annotated with both

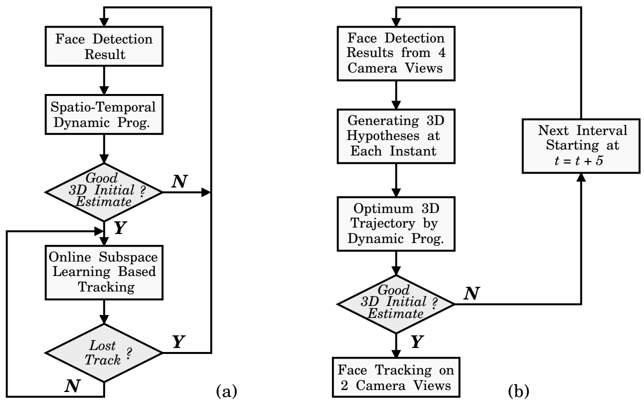
Face Detection Result
Spatio-Temporal Dynamic Prog.
Good 3D Initial Estimate ?
N
Y
Online Subspace Learning Based Tracking
Lost Track ?
Y
N
(a)
Face Detection Results from 4 Camera Views
Generating 3D Hypotheses at Each Instant
Optimum 3D Trajectory by Dynamic Prog.
Next Interval Starting at $t = t + 5$
Good 3D Initial Estimate ?
N
Y
Face Tracking on 2 Camera Views
(b)

**Fig. 3**. *Block diagram of the developed multi-camera 3D head tracking system. (a) Overview; (b) Initialization.*

2D visible face bounding boxes and, by triangulation, 3D head positions [2,5]. Details of the database and available labels are discussed in Section 6.1. Here, it suffices to mention that such data allow the training of statistical face models and to deduce constraints (or even models) of the lecturer's movements.

### 3.2. Overview of the 3D Head Localization Subsystem

The overview diagram of the developed 3D head tracking system is given in Fig. 3(a). It basically consists of an initialization and a tracking component, with tracking drift detection controlling the switch between these two modes. For its initialization, multi-pose face detectors are first applied to all four camera views in the smart room (also referred to in this work as a "quad-frame" – see Fig. 2). Details are provided in Section 4.4. Subsequently, spatio-temporal information of the face detection results over ten consecutive quad-frames is integrated within a dynamic programming (DP) framework, to provide robust initialization. Details are described in Section 4.1 (see also Fig. 3(b)). If the optimal DP trajectory is accepted as a true object, a 2D tracking component kicks in, operating independently in two only camera views, which are selected among the four available views based on the DP result. Details of the tracking algorithm, which is based on online adaptive subspace learning, are presented in Section 4.2. Notice that as long as the DP trajectory is not acceptable, the initialization process is repeated with a shift of five frames. In such a case, a default or no 3D location can be returned. Finally, an important aspect of the system is the re-initialization decision, or equivalently, the drift detection. This is described in Section 4.3, and it is based on a combination of local face detection and calibration-based triangulation to test the consistency of independent tracking in the two camera views (selected based on the DP results).

### 3.3. Overview of the 2D Face Localization Subsystem

In the developed system, 2D face localization is performed based on the 3D head tracking result. Such result provides the approximate region within the 2D frame views, where a visible face could be present, in the following manner: As mentioned above (and further explained in Sections 4.1 and 4.2), the 3D head tracking system uses 2D subspace tracking on two only camera views, selected based on the algorithm initialization stage. For these two camera views, the expected face location is therefore immediately available. For the remaining two camera views, the system considers the projection of the 3D head position estimate (by employing camera calibration information) to obtain an estimate of the head's 2D location in the image frames.

Following this step, multi-pose face detection (see Section 4.4) is applied around the estimated head center in each camera view. If the face detector locates a face, this is accepted. If there is no face detection result, then one of the following two cases occurs: (a) If the camera view in question is one of the two views that have been used in tracking at that instant, the raw 2D tracking result (i.e., the tracked face box) is returned as the face detection output. (b) If however the camera is not a 2D tracking view, no face output is produced. The above face detection strategy has been selected after conducting a number of experiments on the CHIL development data, as described in Section 6.5.

## 4. SYSTEM COMPONENTS

We now proceed to describe the developed system's main algorithmic components in detail. In particular, we first concentrate on the initialization, tracking and drift detection modules of the 3D head tracking subsystem, as depicted in Fig. 3(a). In addition to these, prominent throughout the system is the face detection module, that plays a crucial role in initialization and drift detection, as well as in the 2D localization subsystem discussed in Section 3.3. Its presentation is deferred to the end of this section.

Throughout the algorithmic overview, the following notation will be used: $H^{(t)}$ will denote a hypothesis at instant $t$ concerning the lecturer's head centroid location in 3D world coordinates $(x_t, y_t, z_t)$. Similarly, $h_c^{(t)}$ will represent a hypothesized "visible face" at instant $t$ in camera view $c \in \mathcal{C}$, where $\mathcal{C}$ is the set of available cameras (here, four). Face hypothesis $h$ contains 2D information about the face bounding box, $(u, v, \Delta u, \Delta v)$, namely 2D center coordinates, height, and width. The collection of pixels within it will be denoted by $\mathbf{h}$.

### 4.1. Spatio-Temporal 3D Initialization

Robust initialization is a crucial component in every tracking scheme. In the proposed system, initialization is driven by the face detection module described in detail in Section 4.4. In particular, trained Adaboost-like multi-pose face detectors are applied on all four camera views (over the entire quad-frame) and over all time instants during the initialization phase. However, the resulting detected faces prove insufficient to lead to robust 3D initialization by triangulation alone [46]. This is due to high rates of false positives and missed faces, as discussed in Section 4.4 and quantified in the experiments (Section 6.5) — see also Fig. 4.

Given the challenging nature of face detection in the CHIL scenario, the developed system seeks to utilize additional information, in the form of temporal (video sequences) and spatial (multiple camera views) context. The resulting algorithm integrates both temporal and spatial information from frame-level face detection results into a *dynamic programming* (DP) framework, schematically depicted in Fig. 3(b). In summary, following face detection, 3D hypotheses of the presenter's head location are generated using the calibration information, based on the spatial consistency of the detection result from different camera views. Then, DP applied on the results over ten consecutive quad-frames is used to search for the optimal trajectory of the presenter's head centroid in the 3D space, based on appropriately defined penalty functions. If the optimal trajectory is accepted compared to a threshold, the result is fed into the tracking component described in Section 4.2; otherwise the process is iterated

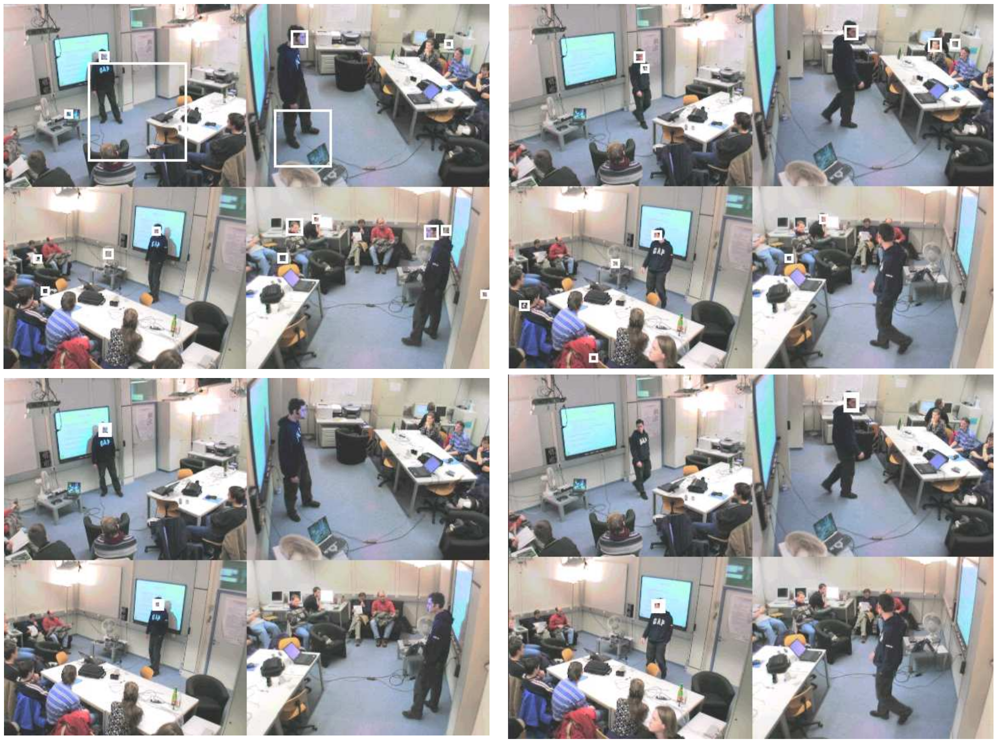

**Fig. 4**. *Spatio-temporal face detection depicted at two instants, for all four camera views.* Upper-row: *Based on frame-level FloatBoost face detection, with no spatial and temporal information utilized.* Lower-row: *After the proposed dynamic programming. Notice that in the latter case, single faces (frontal or profile) are only depicted for the two selected camera views that correspond to the optimal hypothesis.*

with a five frame shift until an acceptable trajectory is determined. An example of the proposed spatio-temporal initialization scheme applied on CHIL lecture data is depicted in Fig. 4. Details of the implementation follow.

### *4.1.1. Generating 3D Hypotheses*

Assuming $n_i$ face detections per camera view, there could be up to

$$\frac{1}{2} \sum_{i,j:i\neq j} n_i \times n_j$$

candidate 3D head locations at each time instant, obtained via pairwise triangulation of detected face bounding box centers, using for example the direct linear transformation (DLT) method [46]. A few of these hypotheses can be readily rejected, for example when large inter-ray distances of the 2D-to-3D maps are observed, or based on collection-site specific spatial constraints. The latter can be learned from development data, and are imposed to distinguish the lecturer from audience members (see also Fig. 1). These constraints result in about half of the room floor surface being allowable for the presenter's ($x$,$y$) location, whereas a 400 mm height range (1500 to 1900 mm) is imposed on the $z$-axis location coordinate. As a result of this process, multiple 3D hypotheses

$$H_i^{(t)} = \text{DLT}\,(h_{k_i}^{(t)}, h_{l_i}^{(t)}) \qquad (1)$$

are generated at every time instant $t$, where indexes $k_i$, $l_i$ specify the face hypotheses in two camera views that yield $H_i^{(t)}$. Hence in this framework, each $H_i^{(t)}$ contains not only the 3D location coordinates of the hypothesized head centroid, but also indexing information about the two camera views that generated it.

### *4.1.2. Trajectories of 3D Hypotheses*

Following generation of a pool of 3D head centroid hypotheses at each time instant $t$, the next step is to perform dynamic programming over the temporal window of interest, in order to obtain the optimal temporal sequence (path or trajectory) of 3D location hypotheses. For this purpose, two main 3D-path cost components are employed. One is a traditional transition cost that penalizes path discontinuities over time. An additional local cost complements it, based on a similarity measure of the 3D hypothesis. This is introduced to reward consistency among the face detection results that generated the hypothesis via (1). As a result, a path

$$\mathbf{H} = \{H_{i_1}^{(t_1)}, H_{i_2}^{(t_2)}, ..., H_{i_n}^{(t_n)}\} \qquad (2)$$

based on $n$ 3D hypotheses of head centroids at times $t_1 < t_2 < ... < t_n$ has a trajectory cost associated to it, given by

$$C^{(t)}(\mathbf{H}) = t_1\, C_B + (t - t_n)\, C_E + \sum_{k=1}^{n-1}(t_{k+1} - t_k)\, C_I + \sum_{k=1}^{n-1} C_T(H_{i_{k+1}}^{(t_{k+1})} |\, H_{i_k}^{(t_k)}) + \sum_{k=1}^{n} C_L(H_{i_k}^{(t_k)}) \;, \qquad (3)$$

over time interval $[\,0\,,\,t\,]$, where $t \geq t_n$ . In (3), $C_T(\bullet|\bullet)$ and $C_L(\bullet)$ denote the transition and local similarity costs, respectively. In addition to those, three constant costs are introduced to account for missing 3D hypotheses or to allow skipping unreliable ones (by essentially duplicating a prior hypothesis) in some of the instants over the temporal window of interest. The three costs, $C_B$, $C_I$, $C_E$ are used for this purpose at the beginning, intermediate, or ending part of the trajectory, respectively. Additional details of the components in (3), as well as the hypothesis search follow.

#### *4.1.3. Local Similarity Cost*

This is used to evaluate the hypothesis at the current instant on the basis of the available camera views that generated it via (1), exploiting spatial information by means of local appearance. The assumption is that if the candidate hypothesis corresponds to an actual 3D object, then the corresponding face regions in the two camera views should have similar color histograms. The cost computation is based on the Bhattacharyya coefficient, and is defined as (see also (1))

$$C_L(H_i^{(t)}) \;=\; -\,\alpha \sum_{b=1}^{m} \sqrt{p_b(\mathbf{h}_{k_i}^{(t)})\, p_b(\mathbf{h}_{l_i}^{(t)})}\;, \tag{4}$$

where $\{p_b(\mathbf{h}) : b = 1, ..., m\}$ denotes the $m$-bin color histogram, based on the face candidate pixel values $\mathbf{h}$, and $\alpha$ is a scalar value used in order to balance the contributions of (4) and (5) in (3).

In our implementation, $p$ is taken to be the 30-bin histogram of the H component of the color HSV space. Furthermore, and in order to improve robustness, the face candidate regions in the computation of (4) are extended: Histograms are computed over rectangles taken to be approximately double (in height only) the detected face bounding boxes $h_{k_i}^{(t)}$ and $h_{l_i}^{(t)}$.

#### *4.1.4. Transition Cost*

The transition cost exploits temporal information, and it is used to penalize non-smooth trajectories, based on the 3D distance between temporally consecutive hypotheses. The cost is specified using Gaussian diffusion, computed between 3D hypotheses $H_i^{(t)}$ and $H_j^{(t-1)}$, as

$$\begin{aligned} C_T(H_i^{(t)}|H_j^{(t-1)}) \;=\; & \frac{1}{2}\log|\Sigma| + \frac{3}{2}\log 2\pi \\ & + (H_i^{(t)} - H_j^{(t-1)})^T \Sigma^{-1} (H_i^{(t)} - H_j^{(t-1)})\,. \end{aligned} \tag{5}$$

In our system, the covariance matrix $\Sigma$ is set to diagonal matrix (100,100,100), assuming that 3D hypothesis coordinates are in mm.

#### *4.1.5. Hypothesis Search*

The searching scheme employs the standard dynamic programming approach, based on cost equation (3) — but with a few twists to better adapt to the task at hand. Available at a given instant $t$ are a pool of local hypotheses $H_i^{(t)}$, $i = 1, ..., m$, and the active trajectories up to $t-1$, which we denote by $\mathbf{H}_j^{(t-1)}$, $j = 1, ..., n$, extending the notation in (2). The latter are accompanied by scores $g_j^{(t-1)}$ that specify the trajectory cost up to $t-1$, based on (3). Then, the active hypotheses at $t$ are obtained as $\mathbf{H}_i^{(t)} = \{\mathbf{H}_{\hat{\jmath}(i)}^{(t-1)}, H_i^{(t)}\}$, where

$$\hat{\jmath}(i) = \underset{j=1,\dots,n}{\arg\min} \{\, g_j^{(t-1)} + C_T(H_i^{(t)}|\, H_j^{(t-1)}) + C_L(H_i^{(t)}) \,\}\,,$$

with the new score $g_i^{(t)}$ being the optimal value of the above minimized expression. In addition to the updated trajectories, active hypotheses $\mathbf{H}_j^{(t-1)}$ may remain "alive" as $\mathbf{H}_j^{(t)} = \{\mathbf{H}_j^{(t-1)}, H_j^{(t-1)}\}$ (slight notation abuse) with a constant penalty $C_I$ added to their score (see (3)). To speed up computations, pruning is performed among the resulting pool of paths, by allowing at most six trajectories to be kept active at any instant $t$. Furthermore, the scheme is terminated at the $10^{th}$ quad video frame ($t_{end} = t_{init} + 10$), with the global optimal trajectory obtained by choosing the active hypothesis with the minimum score at $t = t_{end}$.

In addition, a maximum acceptable score is defined, providing a mechanism to reject the final hypothesis (and hence trigger a new search) if its total cost exceeds a fixed threshold. This threshold, as well as parameters $C_I = C_B = C_E$ and $\alpha$ in (3) and (4), are tweaked empirically, based on detection and false alarm rates on CHIL development data. In the case that the optimal trajectory is rejected, a five quad-frame shift is performed and the search re-initialized. The returned optimal trajectory defines the two camera views on which 2D tracking is to commence, as discussed next.

### 4.2. Adaptive Subspace 2D Tracking

Following successful initialization, a 3D hypothesis is obtained as the last element of the optimal (minimum score) spatio-temporal path at time instant $t_o \doteq t_{end}$. This hypothesis, denoted by

$$H^{(t_o)} \;=\; \mathrm{DLT}\;(\,h_{c'}^{(t_o)}, h_{c''}^{(t_o)})\,,$$

contains the two face detection results and the indexing information of the two camera views, $c', c'' \in \mathcal{C}$, that generated it. Such information allows the tracking phase of the algorithm to commence. This stage consists of two separate 2D tracking processes, running independently and in parallel for each of these two camera views. The 2D processes are based on an adaptive PCA subspace approach that tracks the face bounding box within the single-camera frame sequence. Therefore, at each time instant $t > t_o$, the two trackers generate face bounding boxes $h_c^{(t)}$, $c \in \{c', c''\}$. The 3D head centroid location can then be easily obtained via triangulation as $H^{(t)} = \mathrm{DLT}\;(\,h_{c'}^{(t)}, h_{c''}^{(t)})$, assuming that no tracking drift is detected (see Section 4.3).

The motivation behind this scheme is to reduce computations by tracking using the bare minimum of camera views (two), sufficient for 3D triangulation, but also to do so in the specific views where visible faces (frontal or profile) are expected. Such views contain more discriminating information, as opposed to views that capture the back of the lecturer's head. In addition, they enable the verification of whether the hypothesized tracked object is indeed a visible face, by applying a face detector in its region. This is crucial in detecting possible tracking problems (see Section 4.3). Furthermore, the 2D tracking results may readily provide desired 2D face information in the camera views in question, as discussed in Section 3.3).

At the heart of the proposed scheme lies the 2D PCA subspace tracking approach. As discussed in Sections 1 and 2, adaptability of the subspace to the observed conditions is crucial in improving tracking robustness in the dynamic CHIL scenario, mainly due to variations in head-pose and lighting. Such approaches have already been proposed in the literature, for example in [24]. There, when a new observation is obtained, the PCA subspace is updated to take into consideration the variance contributed by the new observation. However, the method does not provide an updating algorithm for eliminating past observations during tracking. This poses a problem when tracking objects over long durations, since the noise introduced during tracking eventually could bias the PCA subspace away from the characteristic appearance of the desired tracked object. In [59], an $L_\infty$ norm subspace is fitted to the past frames incrementally by Gramm-Schmitt orthogonalization. Though the subspace with $L_\infty$ norm has the advantage of timely incorporating observation novelties into the subspace representation [59], it runs the risk of tracking drift due to its lack of robustness to noise and outliers. PCA on the other hand offers freedom to perform dimensionality reduction and thus ignore tracking noise and assist outlier rejection based on reconstruction error [42]. Therefore, the proposed system adopts the incremental PCA subspace learning approach. In particular, Hall's mechanism [60] is employed to incrementally update the PCA subspace given new observations. In addition, our proposed system also allows subspace adjustment, by eliminating distant past observations in the subspace. This introduces a forgetting mechanism that is absent in Lim's approach [24].

The proposed 2D adaptive subspace tracking scheme consists of three steps, at each time instant (frame) $t$, as discussed next. The presentation refers to faces, but of course the scheme is more general.

(a) *Localization*: The first step is to estimate the new face location at instant $t$, $h^{(t)}$, based on the prior face location, $h^{(t-1)}$, and the available PCA subspace of face appearance at $t-1$ (for simplicity, we drop the camera index in the notation). Let us denote the current PCA subspace by $(\bar{\mathbf{h}}^{(t-1)}, U^{(t-1)}, \Lambda^{(t-1)}, N^{(t-1)})$, with its elements representing, respectively, the mean vector of face appearances, the matrices of retained eigenvectors and eigenvalues, and the current number of observations modeled. The new face location at $t$ is then obtained as

$$h^{(t)} = \arg\min_{h\,\in\,\mathcal{N}(h^{(t-1)})} \|(\,\mathbf{h}-\bar{\mathbf{h}}^{(t-1)}) - U^{(t-1)}U^{(t-1)\,T}(\,\mathbf{h}-\bar{\mathbf{h}}^{(t-1)})\,\|_2 \,, \tag{6}$$

where the minimization occurs over a set of candidate face bounding boxes in the "neighborhood" $\mathcal{N}(h^{(t-1)})$ of the previous face. Note that in (6), the mimized functional corresponds to the distance from the PCA space of the vectors of candidate face pixels, $\mathbf{h}$, within the corresponding face bounding boxes $h$.

(b) *New sample inclusion into subspace*: Once the new face "observation" $h^{(t)}$ becomes available, it's pixel values vector $\mathbf{h}^{(t)}$ gets recruited into the PCA subspace. The subspace can be adapted in an incremental fashion, as described in Alg. 1 of Fig. 5, thus avoiding recomputing the subspace from all its samples.

(c) *Old sample exclusion from subspace*: Following inclusion of the new observation, the PCA subspace receives a second update by excluding a past distant observation vector $\mathbf{h}^{(t-m)}$. This forgetting mechanism is performed as described in Alg. 2 of Fig. 5, avoiding recalculation of the entire subspace. Notice that in contrast to step (b), the process occurs only once the subspace reaches its "steady state" of containing $N^{(t)} = m$ samples, or equivalently for $t \geq t_o + m$.

In our particular implementation, the proposed system employs the most recent $m = 50$ frame observations to construct the PCA subspace. Hence, following tracking initialization, the forgetting mechanism does not commence until after 50 frames are observed. For this initial duration, the algorithm remains identical to [24]. The learned subspace has a dimensionality of up to 15, down from a normalized 20×20-pixel data "template" (the un-normalized template size depends on the detected face at the end of the initialization step). Finally, the optimization in (6) occurs over 169 candidate faces of constant size (equal to the detected face size at initialization), with their centers located at equally spaced points within a square four times in size of the initialized face actual size. Notice therefore that the tracking occurs in constant scale, with only the face location sought.

### 4.3. Tracking Drift Detection in 3D

An important aspect of the system is the re-initialization decision, or equivalently, tracking drift detection on basis of the 2D independent tracking results in the two selected camera views. This is based on a combination of local face detection and calibration-based triangulation to test the consistency of the two tracks at the given time. In more detail, if the inter-ray distance of the two 2D-to-3D mapping rays is larger than a predetermined threshold, this indicates that the two tracked results are inconsistent, hence immediately prompting re-initialization. Furthermore, at each frame, the multi-pose face detectors of Section 4.4 are also applied around the two tracking results to determine whether there indeed exists a face object in the local regions of interest (for example, in the proposed system, this is set to a 80×80 pixel region when running on CHIL seminar data collected at UKA). If faces could not be detected in the local region for several frames (30 in our case) in any of the two camera views, a re-initialization decision is prompted.

```
Alg. 1: INCLUDE $(\bar{\mathbf{h}}^{(t-1)}, U^{(t-1)}, \Lambda^{(t-1)}, N^{(t-1)}, \mathbf{h}^{(t)})$
  $N^{(t)} = N^{(t-1)} + 1$
  $\bar{\mathbf{h}}^{(t)} = \frac{\bar{\mathbf{h}}^{(t-1)} N^{(t-1)} + \mathbf{h}^{(t)}}{N^{(t)}}$
  $d = \bar{\mathbf{h}}^{(t-1)} - \mathbf{h}^{(t)}$
  $g = U^{(t-1)\,T} d$
  $z = d - Ug$
  if $\|z\| = 0$
  $A = \Lambda^{(t-1)} \frac{N^{(t-1)}}{N^{(t)}} + gg^T \frac{N^{(t-1)}}{N^{(t)\,2}}$
   else
  $\begin{cases} v = z/\|z\| \\ r = v^T d \\ A = \begin{pmatrix} \Lambda^{(t-1)} & 0 \\ 0 & 0 \end{pmatrix} \frac{N^{(t-1)}}{N^{(t)}} \\ \quad + \begin{pmatrix} gg^T & gr^T \\ rg^T & rr^T \end{pmatrix} \frac{N^{(t-1)}}{N^{(t)\,2}} \end{cases}$
  $\Lambda^{(t)} = eigenvalue(A)$
  $R = eigenvector(A)$
  $U^{(t)} = [U^{(t-1)}\, v]R$
  return $(\bar{\mathbf{h}}^{(t)}, U^{(t)}, \Lambda^{(t)}, N^{(t)})$
```

```
Alg. 2: EXCLUDE $(\bar{\mathbf{h}}^{(t-1)}, U^{(t-1)}, \Lambda^{(t-1)}, N^{(t-1)}, \mathbf{h}^{(t-m)})$
  $N^{(t)} = N^{(t-1)} - 1$
  $\bar{\mathbf{h}}^{(t)} = \frac{\bar{\mathbf{h}}^{(t-1)} N^{(t-1)} - \mathbf{h}^{(t-m)}}{N^{(t)}}$
  $d = \bar{\mathbf{h}}^{(t-1)} - \mathbf{h}^{(t-m)}$
  $g = U^{(t-1)\,T} d$
  $A = \Lambda^{(t-1)} \frac{N^{(t-1)}}{N^{(t)}} - \frac{gg^T}{N^{(t-1)}}$
  $\Lambda^{(t)} = eigenvalue(A)$
  $R = eigenvector(A)$
  $U^{(t)} = U^{(t-1)} R$
  Prune the subspace bases in $U^{(t)}$ with eigenvalue too
    small in $\Lambda^{(t)}$
  return $(\bar{\mathbf{h}}^{(t)}, U^{(t)}, \Lambda^{(t)}, N^{(t)})$
```

**Fig. 5**. *Brief overview of the incremental adaptive subspace update used for 2D tracking, when including a novel observation (Alg. 1), or excluding a distant past observation (Alg. 2) from the subspace.*

### 4.4. Multi-Pose 2D Face Detection

Face detection is a critical component of the developed system, being used at the initialization (Section 4.1) and drift detection stages (Section 4.3) of the 3D head tracking sub-system, and in addition being the required step to produce 2D face results, based on the 3D head location estimate, as discussed in Section 3.3. Our system adopts a multi-pose face detector approach, with classifiers trained using the FloatBoost technique [38], an AdaBoost variant [33].

#### *4.4.1. AdaBoost and FloatBoost Learning*

AdaBoost provides a simple yet effective approach for stagewise learning of a nonlinear classification function [61]. While a good classifier is difficult to obtain at once, AdaBoost learns a sequence of more easily attainable "weak" classifiers, whose performances may be poor, but better than random guessing. It then boosts (combines) them into a "strong" classifier of higher accuracy.

Viola and Jones [33] successfully applied AdaBoost classification to the face detection problem, following earlier work [62, 63]. There, AdaBoost is adapted to solve three issues: (i) Learning effective features from a large feature set; (ii) Constructing weak classifiers, each based on one of the selected features; and (iii) Boosting the weak classifiers into a stronger one. In the particular two-class face detection problem, tens of thousands of simple Haar wavelet-like features are defined, and an appropriate scheme for their selection is designed. The process is carried out sequentially, at each step $m$ selecting a weak classifier $f_m(\mathbf{h})$, simply designed based on its corresponding feature, over the pool of available features. The weak classifier is added into a linear combination of the already chosen weak classifiers in previous steps, resulting to a stronger one, $F_m(\mathbf{h})$ . The selection of $f_m(\mathbf{h})$ is based on minimizing the classification error of $F_m(\mathbf{h})$ on an appropriately weighted epoch of the training data. The scheme therefore represents a greedy sequential forward search procedure.

An alternative training algorithm, applied to the face detection problem, appears in [38]. This employs the sequential floating search method [64] that allows feature deletion and controlled backtracking during the strong classifier learning process. In particular, a "conditional exclusion" step is added to AdaBoost training. In it, each of the weak classifiers $f_k(\mathbf{h})$ , $0 \leq k \leq m$ , that constitute elements of $F_m(\mathbf{h})$ is examined to check whether removing it may actually reduce classification error of the remaining linear combination. If such situation occurs, and assuming that weak classifier $f_n(\mathbf{h})$ is the one that reduces the error the most when removed, $f_n(\mathbf{h})$ will be deleted, and all classifiers $f_k(\mathbf{h})$, $n < k \leq m$ , will be re-learned. The process results in more expensive training compared to the traditional AdaBoost scheme, but yields more compact sets of weak classifiers for testing.

Both AdaBoost and FloatBoost learning approaches discussed can be used to combine the successively stronger classifiers into a cascade structure [33, 38]. The goal is for the resulting classification structure to quickly reject uninteresting non-face candidates $\mathbf{h}$, while focusing more attention to candidates that appear to be face-like (or confused as such). A simple framework is proposed for this purpose in [33].

#### *4.4.2. Implementation Details*

In our implementation, we use the FloatBoost approach [38] to train cascaded (layered) face classifiers using Haar wavelet features [33]. In particular, since faces may be visible in the available camera views with different head poses, we train two detectors, based on clustering visible faces into two groups: Frontal ones that also contain near-frontal faces, and left-side profile ones pooled together with mirrored right-side profile faces. The two face detectors are trained on development set data, on images cropped based on the available CHIL corpus annotations (see also Section 6.1). For negative examples (non-faces), training samples are cropped from an image database that does not include faces, as well as non-face regions of CHIL corpus frames. Separate face detectors have been trained for each of the three parts of the CHIL database, discussed in Section 6.1. For example, for the "CHIL04" set, 1606 frontal and 1542 profile images have been used. Following FloatBoost training, the resulting frontal face detector consists of 15 layers and 576 Haar wavelet features, whereas the profile view one consists of 30 layers and 4330 features. Notice that during the testing phase, an additional detector of right-side profile view faces is used. This is readily obtaining by mirroring the left-side profile view face detector [38]. An example of detected faces on CHIL data is depicted in the upper rows of Fig. 4.

## 5. ALTERNATIVE 3D TRACKING SYSTEMS

In order to evaluate the performance of the proposed system, we compare it with a number of alternative 3D tracking approaches in experiments reported in Section 6. Two of the systems are only slight variations of the proposed theme; therefore, they are briefly described together with our experiments (see Section 6.3). The remaining two however depart significantly from it. The first, somewhat less so, since it is also face detection based, retaining the same multi-pose FloatBoost detectors and employing an identical drift detection mechanism [23]. However, in contrast to the proposed approach, it relies on detecting lecturer motion in its initialization phase and employs 2D mean shift tracking [40]. The second system considered is based on background subtraction for tracking and constitutes an adaptation of the IBM "Smart Surveillance Engine" [25, 65] to the CHIL tracking task. The two systems are described next.

### 5.1. Motion and Mean-Shift Tracking Based System

Similarly to the proposed system, this alternative approach consists of three components, namely 3D initialization, 2D tracking, and drift detection. The latter, as well as the face detection part of the initialization component are identical to the proposed system, as described in Sections 4.3 and 4.4, respectively. However, the system lacks the more sophisticated spatio-temporal dynamic programming framework for initialization, using instead a motion detection based approach to identify candidate regions for initialization. In addition, it replaces adaptive subspace tracking with the mean-shift tracking algorithm. The two components that differ from the proposed system are briefly discussed below. More information can be found in [23]. Notice that as with the approach in Section 3.3, this system can be used for tracking "visible" faces.

#### *5.1.1. Initialization*

At the initialization stage, three primary modules are employed: Motion analysis, face detection, and triangulation (based on camera calibration information).

First, independently for each camera view, motion history is estimated to rapidly determine where movement has occurred. The algorithm used is based on work by Davis and Bobick [66]. Obtaining a foreground silhouette is achieved through subtraction between two consecutive frames instead of background subtraction. As the

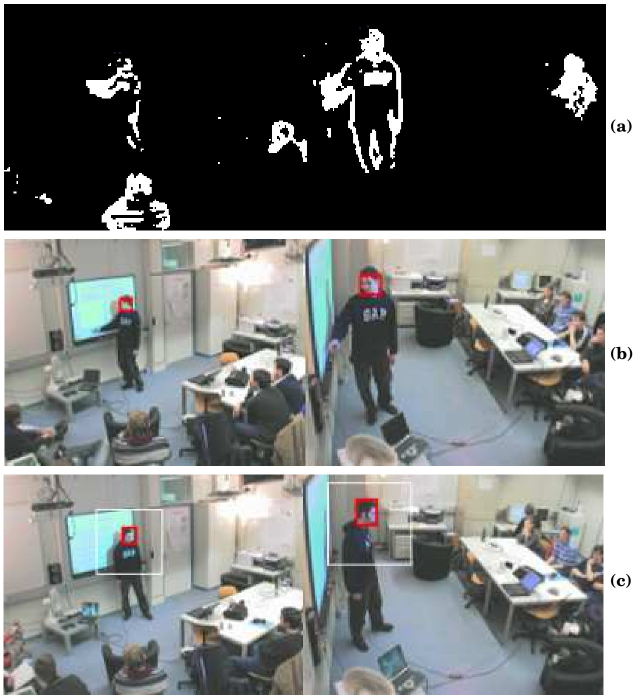

**Fig. 6**. *Examples of processing steps in an alternative 3D head tracking system, based on face detection, motion estimation, and mean shift tracking [23]. (a) Motion history image for two camera views; motion objects are segmented as foreground (white pixels). (b) Multi-pose face detection result, after FloatBoost face detectors are applied locally around the resulting foreground region. (c) Local face detection applied within windows around the mean shift based tracking results in the two camera views.*

person moves, the most recent foreground silhouette is copied as the highest value in the so-called "motion history image" (MHI). MHI pixel values that fall below a threshold are set to zero. An example of the algorithm applied to two camera views is depicted in Fig. 6(a).

Subsequently, a multi-pose face detector, identical to the one of the proposed system (Section 4.4), is applied to the foreground region only (where motion occurred), instead of the whole frame. The detection results for each camera view can then be used to verify whether the detected faces belong to the same person, based on calibration information [46], thus providing the 3D head position. The highest lying 3D position within the general seminar presenter area (see also Fig. 1) is returned as the initialization estimate for subsequent tracking.

The above algorithm could in principle be applied to all four camera views. However, in order to reduce the pool of 3D initialization candidates, two only camera views are being used in the implementation of [23], as many more candidates would have arisen from four versus two views. These cameras have been selected based on development data from each lecture as the cameras with the highest percentage of (near-)frontal faces. This is possible for "CHIL03" and "CHIL04" data, where development and evaluation sets are available for each of the lectures in the corpus (see Section 6.1), and assumes that the lecturer's general location behavior would not change over the duration of the seminar. Alternative camera selection schemes can however be easily devised.

### *5.1.2. Mean Shift Tracking*

Following the initialization component and the successful location of the presenter's face, the algorithm switches into its tracking mode. A color-based face model of the detected face region is first created for tracking in each of the two camera views. In particular, the one-dimensional histogram of the H component in the HSV color space is used for this purpose. The mean shift iteration algorithm is then employed for tracking [40], based on the Bhattacharyya coefficient, around a target position predicted by means of Kalman filtering [67]. The algorithm is applied separately in the two camera view images to find the best target candidate. Subsequently, triangulation provides the 3D position estimate, with drift detection, as in Section 4.3, flagging possible inconsistencies that trigger re-initialization.

## 5.2. Background Subtraction Based System

This system constitutes a 3D tracker, developed on top of the IBM "Smart Surveillance Engine" (SSE) 2D (image plane) tracker [68]. The system applies the 2D tracker independently to each of the four available camera views, and then integrates the information in 3D [69].

The 2D component is based around a background-subtraction object detection system [25, 65], which uses a multiple Gaussian color model at each pixel. Objects are tracked in the image plane from frame to frame using the "ColourField" tracking method described in [68]. A preliminary extension of this system to 3D tracking, called the "Face Cataloger" appears in [9]. There, the 2D tracker was applied independently in two, nearly orthogonal, views, and used a head detection algorithm to locate the head center, regardless of pose. The estimated head points from pairs of tracks in the 2D views were then triangulated to determine correspondence and estimate 3D head centroid positions. Subsequently, this information was used to steer a calibrated pan-tilt-zoom camera to the head position to allow head close-ups to be captured. An additional face detection system guided a closed-loop control system to further zoom in, if the face was visible from the steered camera, overcoming localization and calibration errors in the fixed cameras.

Since then, improvements in the underlying 2D tracking algorithm allowed a new 3D tracking algorithm to be developed for the CHIL task [69]. This approach dispensed with the head detector, which had limitations when multiple targets were being tracked, and was found to be unnecessary in lectures, where the head is almost always the highest point of the presenter's body. In this version of the tracker, the underlying improved 2D tracking algorithms of the IBM SSE system are again employed, unmodified from their usual outdoor surveillance configuration. The 2D tracker provides a temporally-smoothed model of the objects observed in each view, together with each object's location, tracked through occlusions. The 2D track information however is not used in the 3D engine; instead, temporal consistency is applied directly in 3D.

In more detail, at each frame, the 2D tracker is applied, and the resulting 2D probabilistic models are used to determine the position of the head top. This is taken to be the point whose $y$ coordinate is the top of the object model bounding box and whose $x$ coordinate is that of the centroid of the upper sixth of the model. This assumes that people's projections in the camera views have the head uppermost, but are not necessarily vertical.

The resulting 2D object points are considered as hypotheses for the top of the speaker head, and when coupled with the camera calibration information, each gives a 3D ray, along which the speaker's head might lie. Validation for these hypotheses in other views is then sought, by computing the shortest distance $d_{r_1,r_2}$ between each pair

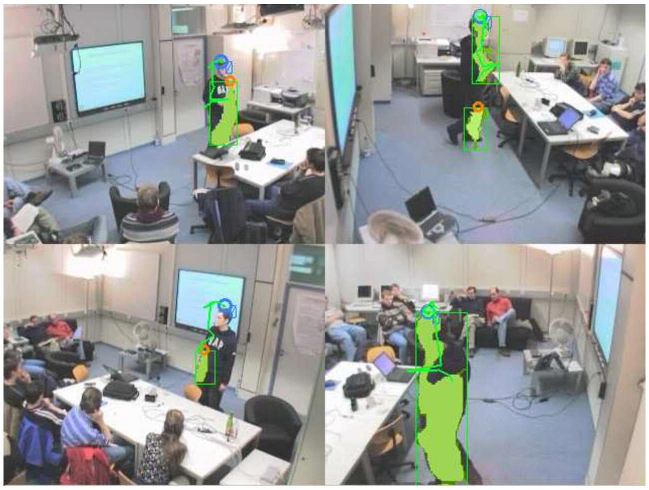

**Fig. 7**. *Detection results for the background subtraction based tracker on four synchronous camera views. Foreground blobs are shown in solid green. Candidate head-top points are depicted as small orange circles. 3D head location hypotheses are shown back-projected as larger blue circles. The current Viterbi path is depicted as a green line.*

$(r_1, r_2)$ of such rays from different cameras. All such pairings are evaluated, sorted and compared to a distance threshold $D$ set to 300 mm, with the closest match being considered first. The following algorithm is used for this purpose:

1. Start with a pool of ray hypotheses for each camera.
2. If $d_{r_1 r_2} < D$, create a 3D hypothesis at the midpoint, $\mathbf{p}_{r_1, r_2}$, of the shortest inter-ray line.
3. Search the remaining cameras for other rays $r_3$ that pass this point within a distance of $D$, move the hypothesis to $\mathbf{p}_{r_1, r_2, r_3, \ldots}$, the least squares fit for all rays, and repeat for any remaining cameras.
4. Store the hypothesis $h_i(t)$, and remove from the pools the rays just used in this 3D hypothesis.
5. Repeat from step 2.

These 3D hypothesis points can be associated over time and concatenated into 3D tracks. For this purpose, dynamic programming is employed to find the best track hypothesis through the temporal sequence of 3D head-top hypotheses. The approach uses a beam search with up to $N$ (typically 50) search hypotheses active, to search for the shortest path passing through head location hypotheses. Trajectory costs are given by (3), but with a few differences; namely, $C_T(H_i^{(t)} | \, H_j^{(t-1)}) = \| H_i^{(t)} - H_j^{(t-1)} \|$ and $C_L(H_i^{(t)}) = 0$. At each time instant, all paths are updated, where each path can be retained with no additional evidence (with a penalty), or by adding one of the 3D location hypotheses for that instant. This produces many search hypotheses which are sorted by cost, with only the top $N$ retained for the next instant. At the end, the lowest cost path is retained as the "best" path through the 3D location hypotheses. Part of this process is depicted in Fig. 7, where background subtraction results on synchronous frames are shown for all four cameras, marked with 2D head-top candidates and the back-projected locations of a 3D hypothesis, as well as the current best track.

To allow effective background subtraction, background images are used when testing this algorithm on the CHIL lecture corpus. These images are derived by splicing frames from the development set together, so as to remove the lecturer. This process is performed automatically, based on development CHIL data, and is possible for the "CHIL03" and "CHIL04" sets, since they contain development and evaluation data from the same lectures (see also Section 6.1). Furthermore, and similarly to all trackers used in this work, spatial constraints about the lecturer's 3D location are utilized to improve performance (see also Fig. 1).

## 6. EXPERIMENTS ON THE CHIL CORPUS

We now proceed to evaluate the performance of the proposed tracking scheme on the CHIL lecture corpus and compare it to alternative approaches. The emphasis in our experiments is placed on the 3D lecturer tracking task, but we also consider the 2D face localization task at the end of the section. Before reporting results, we briefly describe the three parts of the CHIL corpus, its annotations, and the adopted evaluation metrics.

### 6.1. The CHIL Lecture Corpus

Our experiments are conducted on the CHIL database. This consists of three subsets, with a fourth set becoming available in late February 2007.

(i) *CHIL03*: This first dataset was collected in 2003 at the smart room of Universität Karlsruhe, in Germany (UKA), and contains seven lectures, each split into two development and two evaluation segments of approximately five minutes duration each. It therefore consists of 14 development and 14 evaluation segments. This set will be referred to as the "CHIL03" dataset, and it has been used in internal CHIL consortium evaluations during the summer of 2004.

(ii) *CHIL04*: The second phase of data collection took again place at the UKA smart room in late 2004. This effort resulted in five lectures, split in a similar fashion to the "CHIL03" set into ten development and ten evaluation segments, each five minutes in duration. This will be referred to as the "CHIL04" set and has been employed in internal CHIL consortium evaluations in January 2005.

(iii) *CHIL05*: The most recent set is significantly more diverse, containing 18 development and 24 evaluation segments of lectures collected at two smart rooms, one located at UKA and the second at the Istituto Trentino di Cultura (ITC), in Italy (see also Figs. 1 and 2). The development and evaluation sets correspond to disjoint lectures. This set will be referred to as "CHIL05", and it has been used in the first international evaluation campaign on the "Classification of Events, Activities and Relationships" (CLEAR) in March, 2006 [5]. It should also be mentioned that this collection effort includes three additional recording sites, partners of the CHIL consortium, including IBM Research. These data however belong to the so-called "interactive-seminar" (or meeting) scenario, where the aim (in terms of tracking) is to determine the location of all meeting participants, typically being less than six in total. This part has been excluded from our experiments, since we focus on the seminar lecturer tracking task.

All video data in the three sets have been recorded using four synchronous corner cameras at 15 Hz. The frame resolution is 640×480 pixels for the UKA site and 800×600 pixels at ITC. In terms of data annotations, visible face locations have been manually labeled in all frame views for every 1.0s (second) for the "CHIL05" data and 0.67s for the "CHIL03" and "CHIL04" sets. Furthermore, the bounding boxes of such faces have been labeled in the "CHIL04" and "CHIL05" sets, with additional facial feature points (nose bridge and eyes) annotated in the latter. In all cases, the corresponding 3D head centroid location is also given, as derived by triangulating the face labels across camera views. Therefore, evaluation of tracking algorithms is possible at the instants with available ground truths (at 0.67s or 1.0s intervals) using appropriate metrics, as discussed next.

## 6.2. Evaluation Metrics

A number of metrics are used in our experiments to benchmark performance of 3D-head and 2D-face tracking algorithms. All are computed by comparing algorithmic outputs (estimated 3D head centroid locations or face bounding boxes) to their corresponding annotated ground truths on the evaluation data sets. These metrics have evolved over the multi-year duration of CHIL project technology evaluations, as it is explained in more detail in the case of 3D tracking.

During the first two years of CHIL internal evaluations (datasets “CHIL03” and “CHIL04”), the following were used to benchmark 3D tracking performance:

(i) *3D error*: This corresponds to the mean Euclidean 3D distance in millimeters (mm) between the estimated and the ground truth position of the head centroid in 3D coordinates. An additional 3D metric has been deemed of interest, namely the percentage of time instants, where the 3D error is smaller than 300 mm. This is denoted by “% 3D err $< 300$” in Table 1.

(ii) *2D error*: This is the mean Euclidean 2D distance in mm between the projection on the smart room floor of the estimated 3D head center and that of the corresponding ground truth projection. Furthermore, “% 2D err $< 300$” is the percentage of time instants, where the 2D error is smaller than 300 mm.

The above metrics have been modified as part of the CLEAR 2006 evaluation campaign based on the “CHIL05” dataset [5], in order to become harmonized with metrics used in the VACE program research community [70]. In particular, two metrics have been identified as relevant to all tracking evaluations on CHIL data, spanning both single- and multi-person, as well as single- and multi-modal tracking conditions [5, 71]. These are:

(i) *Multiple object tracking accuracy* (MOTA), measured as the percentage (%) of correct correspondences (mappings) of estimated and ground truth persons over the evaluation set of time instants. Of course, in the case of single-person tracking, as is the lecturer tracking task considered here, the mapping problem becomes trivial, since there is at most one hypothesized and one reference person. In such a case, the hypothesis is considered correct when the 2D Euclidean distance between the estimated location and the ground truth (both projected to the smart room floor), as compared to a threshold set to 500 mm. Notice that only 2D distance is considered, although the proposed head tracking system provides 3D location information. It is worth mentioning that the metric penalizes guessing (for example, a default hypothesis). Such a strategy would in most cases result in two errors for each default estimate: a false positive and a miss.

(ii) *Multiple object tracking precision* (MOTP): This is measured in mm, and is simply the average 2D Euclidean distance computed over the correct reference-hypothesis mappings. Its value therefore ranges between 0 and 500 mm. Clearly, the MOTP metric becomes identical to the average 2D error metric discussed earlier, if MOTA reaches 100%.

Finally, for the 2D face detection task, a total of five metrics have been identified by the CHIL consortium for use in the CLEAR 2006 evaluations [5]. Results based on the following three are reported in Section 6.5:

(i) Percentage of *correctly detected faces* (“Corr”), namely the percentage of detected faces with hypothesis–reference face bounding-box centroid distance more than half the size of the reference face.

(ii) Percentage of *wrong face detections* (“Err”), accounting for false positives (this includes detected faces with hypothesis–reference bounding-box centroid distance larger than half the reference face size).

(iii) Percentage of *missed face detections* (“Miss”).

In these metrics, the reference face size is defined as the average of height and width of the annotated bounding box.

**Table 1**. *Comparison of 3D head-tracking performance of various algorithms on the CHIL evaluation sets of 2003 and 2004. Clearly, the proposed system (DPAS) performs best.*

| Data | “CHIL03” | | | | |
|---|---|---|---|---|---|
| Metrics | DPAS | DPAS-f | BGS | MMS | DPAS-d |
| 3D err (mm) | 140.0 | 270.2 | 278.4 | 253.9 | 1649.4 |
| 2D err (mm) | 123.6 | 217.3 | 204.7 | 228.3 | 1230.7 |
| 3D err $< 300$ | 92.9% | 82.5% | 81.2% | 84.6% | 13.2% |
| 2D err $< 300$ | 93.3% | 84.3% | 84.1% | 85.3% | 14.6% |
| Data | “CHIL04” | | | | |
| Metrics | DPAS | DPAS-f | BGS | MMS | DPAS-d |
| 3D err (mm) | 155.2 | 267.4 | 480.3 | 467.4 | 1852.4 |
| 2D err (mm) | 141.8 | 208.9 | 436.9 | 441.1 | 1635.1 |
| 3D err $< 300$ | 95.4% | 83.6% | 47.7% | 78.9% | 10.9% |
| 2D err $< 300$ | 95.6% | 85.7% | 57.1% | 80.7% | 12.6% |

## 6.3. 3D Head Tracking Results on “CHIL03 / 04” Data

In the first set of experiments, we concentrate on the “CHIL03” and “CHIL04” subsets of the corpus. As discussed above, these contain non-overlapping development and evaluation subsets that correspond to the same lectures. This fact allows the training of relatively accurate face detectors, since they cover the same lecturer population (akin to a “multi-subject” training/testing scenario, as opposed to the more challenging “speaker-independent” case).

On these sets, we compare a total of five tracking algorithms:

(i) *DPAS*: This is the proposed face-detection based scheme that uses *dynamic programming* and *adaptive subspace* tracking.

(ii) *DPAS-f*: This is a variation of the proposed scheme, where no *forgetting* mechanism is introduced in the adaptive subspace tracking stage, thus the influence of past distant observations is retained until re-initialization is triggered.

(iii) *DPAS-d*: This is a trivial variation of the proposed scheme, where no *drift detection* is present. The algorithm gets initialized at the beginning of the multi-camera video sequence, and remains in the tracking stage, with no re-initialization until the sequence ends.

(iv) *MMS*: This corresponds to the algorithm described in Section 5.1. It constitutes a face-detection driven approach with *motion* based foreground segmentation and *mean shift* tracking.

(v) *BGS*: This is the system presented in Section 5.2 that uses *back-*

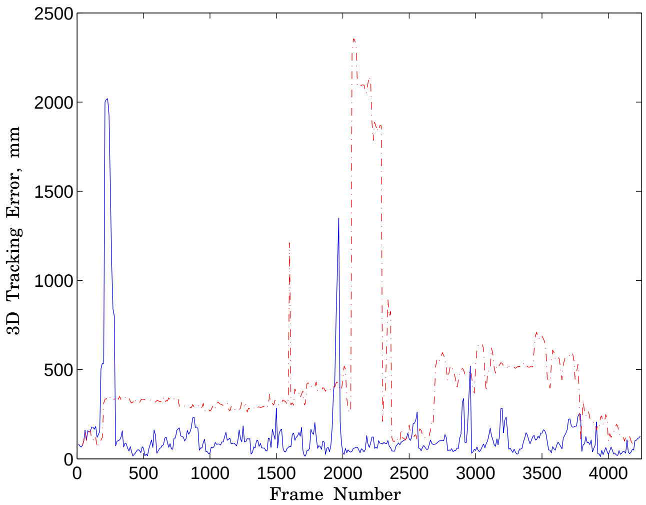

**Fig. 8**. *Typical tracking behavior of the proposed system (DPAS: solid line), compared with its variant (DPAS-f: dashed line) with no forgetting mechanism, evaluated over a CHIL lecture segment.*

*ground subtraction* and dynamic programming.

All above systems are run to always return a 3D head centroid location. In case the algorithm fails to do so (for example, failing to initialize, as discussed in Section 4.1), the returned location defaults to the middle of the presenter's area or the previous estimate in time, if available (DPAS and MMS systems), or an interpolated location between existing estimates immediately before and after the particular instant (BGS system). Concerning face-detection based schemes, development set data are used to train the frontal and profile FloatBoost based face detectors, as discussed in Section 4.4. Furthermore, other system parameters, such as spatial constraints (all methods), DP costs (see for example Section 4.1), inter-ray distance thresholds (e.g., Sections 4.1 and 4.3), and tracking template sizes (Section 4.2, among others) are empirically determined on development data.

Results based on the 3D/2D error metrics discussed in Section 6.2 are depicted in Table 1. It is clear that the proposed system (DPAS) significantly outperforms all others. Interstingly, both systems described in Section 5 (MMS and BGS) achieve similar performance, but exhibit approximately twice (for the "CHIL03" set) or three times (for "CHIL04") the error of the proposed scheme. As expected, the variant of the proposed system, where no drift detection is present (DPAS-d), fails miserably. Finally, it is important to note that the introduction of the forgetting mechanism in adaptive subspace tracking plays a significant role in improving performance. This becomes clear from Table 1, since removing this component (DPAS-f system) almost doubles the tracking error (over DPAS). This is also illustrated in Fig. 8, where the evolution of 3D tracking error over time (quad-frame number) is depicted for one lecture segment.

### 6.4. 3D Head Tracking Results on "CHIL05" Data

We next present the performance of the proposed system (DPAS) on the "CHIL05" set. Given its overwhelmingly better results over the alternative tracking schemes, as demonstrated in Section 6.3, no additional comparisons are reported, with the exception of the DPAS-f variant.

Table 2 presents the summary of the developed 3D head tracking system performance on the "CHIL05" corpus. Results are reported on both development and evaluation sets, listed per collection site, and cumulatively. As depicted in Table 2, the developed system achieves a tracking accuracy of 85.96% on the CLEAR'06 evaluation set, with a tracking precision of 88 mm. Notice that the performance on the development set was significantly worse, at 71.11% MOTA and 99 mm MOTP, due to poor tracking on three development segments. When excluding them, performance on the development set becomes 94.44% MOTA and 90 mm MOTP. Similarly, performance on the evaluation set is unsatisfactory in two segments. Excluding them boosts evaluation set MOTA to 93.00% and MOTP to 86 mm. Notice, that if we were to use the metrics of Section 6.3, the resulting performance would have been an average 2D (3D) error of 139.1 (145.5) mm on the "CHIL05" evaluation sets, which is comparable to the DPAS tracking system performance on the "CHIL03" and "CHIL04" evaluation sets (see Table 1).

**Table 2**. *Performance of the proposed algorithm (DPAS) for 3D head tracking on the "CHIL05" development (DEV) and evaluation (EVA) sets, depicted per collection site and cumulatively. Number of seminar segments are also listed.*

| "CHIL05" Data | | | Metrics | |
|---|---|---|---|---|
| Set | Site | #Sem | MOTA (%) | MOTP (mm) |
| D | ITC | 1 | 21.78 | 148 |
| E | UKA | 18 | 79.47 | 93 |
| V | all | 19 | 71.11 | 99 |
| E | ITC | 2 | 98.33 | 92 |
| V | UKA | 24 | 84.94 | 88 |
| A | all | 26 | 85.96 | 88 |

Two additional results are worth mentioning: As already discussed, the MOTA metric penalizes guessing. This fact has been taken into consideration in the DPAS system: For its evaluation on "CHIL05" data, it returns no 3D hypothesis when initialization fails (see Section 4.1). The exact approach was fine-tuned on the development set, where it boosted the MOTA metric significantly on the 18 UKA segments from an original 69.27% (when always providing a hypothesis) to 79.47%, as reported in Table 2. The second result is a performance comparison between the proposed scheme (DPAS) and its DPAS-f variant, where no forgetting mechanism is present. The latter degrades MOTA to 81.2%; furthermore, it exhibits significantly more tracking drifts, on the average every 193.9 quad-frames (instants), compared to 241.6 of the proposed DPAS tracker.

Additional comparisons of the proposed DPAS scheme with six alternative systems [16–18, 20–22] can be found in [5, pp. 29], as part of the CLEAR 2006 official evaluation. These systems have been briefly overviewed in Section 2.

### 6.5. 2D Face Localization Results on "CHIL05" Data

In the final set of experiments, we report the performance of the proposed 2D face localization subsystem, based on the DPAS head tracking system, as discussed in Section 3.3. The results are reported on the "CHIL05" set, used in the CLEAR evaluation campaign (see also [5, pp. 34]).

A summary of system performance based on the metrics of Section 6.2 is given in Table 3. The system achieved 54.5% correct detections, with 37.2% erroneous detections and 18.9% misses. This

**Table 3**. *Performance of 2D face tracking on the "CHIL05" development (DEV) and evaluation (EVA) sets, depicted per collection site and cumulatively. Number of seminar segments are also listed. All metrics are expressed in %.*

| "CHIL05" Data | | | Metrics (%) | | |
|---|---|---|---|---|---|
| Set | Site | #Sem | Corr | Err | Miss |
| D | ITC | 1 | — | — | — |
| E | UKA | 18 | 74.17 | 21.04 | 15.18 |
| V | all | 19 | — | — | — |
| E | ITC | 2 | 84.75 | 28.70 | 3.14 |
| V | UKA | 24 | 52.64 | 37.68 | 19.89 |
| A | all | 26 | 54.44 | 37.18 | 18.95 |

performance can be considered relatively good, if one takes into account the extremely challenging nature of the task and the rather strict evaluation metrics. In particular, by comparing the UKA development and evaluation set performance in Table 3, one can notice that the performance drops significantly, due to the different lecturer population sets (a purely "speaker independent" evaluation framework is considered). Furthermore, errors and misses are relatively balanced on the development set, but not so on the evaluation data.

A final remark concerns the adopted strategy described in Section 3.3 for face detection. A number of approaches have been considered for producing 2D face results from the 3D head location estimate in an effort to reduce and balance the false positive ("Err") and negative ("Miss") error rates. Among them, an interesting modification of the proposed method is to always return the 2D tracking result on the two selected camera views where the subspace tracking takes place (Section 4.2), and only apply multi-pose face detection to the two non-tracked camera views around a region of interest based on the 3D head estimate. This is in contrast to first applying the multi-pose face detector on all four views, and only resorting to the tracking result of the selected camera views when the detector fails to return a face. The performance of the former approach was measured on seven UKA development set seminars at 77.26% Corr, 18.67% Err, and 9.37% Miss, compared to the superior 85.92% Corr, 9.95% Err, and 9.43% Miss of the adopted approach.

### 6.6. System Run-Time Performance

There has been no particular effort to optimize the proposed system implementation. To reduce face detection overhead and allow speedier development, the whole system has been implemented in a cascade, where face detection is first applied at all instants and all camera views (as in Section 4.4), before feeding its output to the remaining system modules (described in Sections 4.1–4.3). In practice, this is of course suboptimal, as the two 2D tracking processes (Section 4.2) can perform most of the required work in real time – 20 f/s (frames per second) on a P4 2.8 GHz, 512 MByte desktop. In contrast, face detection over the entire frame in four camera views is significantly slower and runs only at about 2 f/s.

## 7. SUMMARY AND DISCUSSION

In this paper, we have presented a vision system for joint 3D head and 2D face tracking for multi-camera smart room settings, where calibrated cameras with wide, overlapping fields of view synchronously record human interaction. In particular, the system has been developed for single-person tracking of the presenter in the CHIL lecture scenario. We described details of the system components, with important highlights being the use of AdaBoost-like multi-pose face detectors, employment of a spatio-temporal dynamic programming algorithm to initialize 3D location hypotheses, and the use of an adaptive subspace learning based 2D tracking scheme with a forgetting mechanism, as a means to reduce tracking drift and increase robustness. The proposed system deviates significantly from other literature work, by not relying on motion estimation, background subtraction, or human body appearance modeling.

We have extensively tested the system on three releases of the CHIL lecture corpus. The proposed system exhibited excellent results with 3D average tracking errors of 140, 155, and 146 mm on three test sets, and outperformed a number of competitive techniques considered in this paper, ranging from simple system variants to entirely different approaches. These experiments, as well as results of the CLEAR 2006 evaluation campaign, demonstrate that the proposed approach is well suited to the problem.

Nevertheless, the system has potential limitations: For example, it is clearly inappropriate for room/camera configurations that consistently result in capturing faces in a resolution too small to allow their detection. A second issue concerns extending the framework to multi-person tracking. Clearly, its 2D tracking and 3D drift detection modules are readily applicable to the multi-person task. However, robust redesign of the initialization module is more challenging. For this purpose, a dynamic programming framework that produces multiple tracks is envisaged, with the number of retained tracks optimized by ad-hoc or information-theoretic approaches.

In future work, we plan to continue research on the topic by working on the multi-person tracking problem. An additional area of interest concerns exploring appropriate multi-camera fusion schemes to allow the system tracking component to directly operate in the 3D space. A more efficient implementation in order to achieve faster run-time performance is also among our goals.

## ACKNOWLEDGEMENTS

This work was partially supported by the European Commission under integrated project CHIL, "Computers in the Human Interaction Loop", contract number 506909. The authors would like to acknowledge contributions by a number of IBM colleagues to the development of the background subtraction tracker used in this paper, through their work on the IBM Smart Surveillance Engine (SSE); in particular, Ruud Bolle, Lisa Brown, Jonathan Connell, Norman Haas, Arun Hampapur, Sharath Pankanti, and Ying-Li Tian. In addition, contributions by UIUC colleagues Jilin Tu and Ming Liu to the proposed system development are also acknowledged. Finally, Aristodemos Pnevmatikakis at the Athens Institute of Technology (AIT), Greece, has kindly provided background images of the "CHIL03" and "CHIL04" data sets that have been used in the testing of the background subtraction based tracker.

## 8. REFERENCES

[1] CHIL: "Computers in the Human Interaction Loop" [Online]. Available: `http://chil.server.de`

[2] D. Mostefa, N. Moreau, K. Choukri, G. Potamianos, S.M. Chu, A. Tyagi, J.R. Casas, J. Turmo, L. Christoforetti, F. Tobia, A. Pnevmatikakis, V. Mylonakis, F. Talantzis, S. Burger, R.

Stiefelhagen, K. Bernardin, and C. Rochet, “The CHIL audiovisual corpus for lecture and meeting analysis inside smart rooms,” [Submitted to:] *Journal of Language Resources and Evaluation*, 2007.

[3] R. Stiefelhagen and J. Garofolo (Eds.), *Multimodal Technologies for Perception of Humans: First International Evaluation Workshop on Classification of Events, Activities, and Relationships, CLEAR 2006*, Springer, LNCS vol. 4122, 2007.

[4] RT06s: “The Rich Transcription 2006 Spring Meeting Recognition Evaluation” [Online]. `http://www.nist.gov/speech/tests/rt/rt2006/spring`

[5] R. Stiefelhagen, K. Bernardin, R. Bowers, J. Garofolo, D. Mostefa, and P. Soundararajan, “The CLEAR 2006 evaluation,” in [3], pp. 1–44, 2007.

[6] A. Stergiou, A. Pnevmatikakis, and L. Polymenakos, “A decision fusion system across time and classifiers for audio-visual person identification,” in [3], pp. 223–232, 2007.

[7] M. Wölfel, K. Nickel, and J. McDonough, “Microphone array driven speech recognition: Influence of localization on the word error rate,” in *Proc. Joint Works. on Multimodal Interaction and Related Machine Learning Algorithms (MLMI)*, LNCS vol. 3869, pp. 320–331, Edinburgh, United Kingdom, 2005.

[8] C. Pinhanez and A. Bobick. “Intelligent studios: Using computer vision to control TV cameras,” in *Proc. Works. on Entertainment and AI/Alife*, pp. 69–76, 1995.

[9] A. Hampapur, S. Pankanti, A.W. Senior, Y.-L. Tian, L. Brown, and R. Bolle, “Face cataloger: Multi-scale imaging for relating identity to location,” in *Proc. IEEE Conf. Advanced Video Signal Based Surveillance*, pp. 13–20, Miami, FL, 2003.

[10] M.N. Wallick, Y. Rui, and L. He, “A portable solution for automatic lecture room camera management,” in *Proc. Int. Conf. Multimedia Expo (ICME)*, 2004.

[11] X. Zhou, R.T. Collins, T. Kanade, and P. Metes, “A master-slave system to acquire biometric imagery of humans at distance,” in *Proc. ACM SIGMM Int. Works. on Video Surveillance*, 2003.

[12] G. Potamianos and P. Lucey, “Audio-visual ASR from multiple views inside smart rooms,” in *Proc. Int. Conf. Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 35–40, Heidelberg, Germany, 2006.

[13] J.-Y. Bouguet, “Camera Calibration Toolbox” [Online]. Available: `http://www.vision.caltech.edu/bouguetj/calib_doc/`.

[14] A. Pnevmatikakis and L. Polymenakos, “2D person tracking using Kalman filtering and adaptive background learning in a feedback loop,” in [3], pp. 151–160, 2007.

[15] M.C. Nechyba and H. Schneiderman, “PittPatt face detection and tracking for the CLEAR 2006 evaluation,” in [3], pp. 161–170, 2007.

[16] N. Katsarakis, G. Souretis, F. Talantzis, A. Pnevmatikakis, and L. Polymenakos, “3D audiovisual person tracking using Kalman filtering and information theory,” in [3], pp. 45–54, 2007.

[17] K. Nickel, T. Gehrig, H.K. Ekenel, J. McDonough, and R. Stiefelhagen, “An audio-visual particle filter for speaker tracking on the CLEAR’06 evaluation dataset,” in [3], pp. 69–80, 2007.

[18] K. Bernardin, T. Gehrig, and R. Stiefelhagen, “Multi- and single view multiperson tracking for smart room environments,” in [3], pp. 81–92, 2007.

[19] K. Nickel, T. Gehrig, R. Stiefelhagen, and J. McDonough, “A joint particle filter for audio-visual speaker tracking,” in *Proc. Int. Conf. Multimodal Interfaces (ICMI)*, Trento, Italy, 2005.

[20] A. Abad, C. Canton-Ferrer, C. Segura, J.L. Landabaso, D. Macho, J.R. Casas, J. Hernando, M. Pardàs, and C. Nadeu, “UPC audio, video and multimodal person tracking systems in the CLEAR evaluation campaign,” in [3], pp. 93–104, 2007.

[21] R. Brunelli, A. Brutti, P. Chippendale, O. Lanz, M. Omologo, P. Svaizer, and F. Tobia, “A generative approach to audio-visual person tracking,” in [3], pp. 55–68, 2007.

[22] B. Wu, V.K. Singh, R. Nevatia, and C.-W. Chu, “Speaker tracking in seminars by human body detection,” in [3], pp. 119–126, 2007.

[23] Z. Zhang, G. Potamianos, A. Senior, S. Chu, and T. Huang, “A joint system for person tracking and face detection,” in *Proc. Int. Work. Human-Computer Interaction (ICCV 2005 Work. on HCI)*, pp. 47–59, Beijing, China, 2005.

[24] J. Lim, D. Ross, R.-S. Lin, and M.-H. Yang, “Incremental learning for visual tracking,” in *Proc. NIPS*, 2004.

[25] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, A. Senior, C.-F. Shu, and Y.-L. Tian, “Smart Video Surveillance,” *IEEE Signal Process. Mag.*, 22(2): 38–51, 2005.

[26] M. Isard and J. MacCormick, “BraMBLe: A Bayesian multiple blob tracker,” in *Proc. Int. Conf. Computer Vision*, vol. 2, pp. 34–41, 2003.

[27] D.M. Gavrila, “The visual analysis of human movement: A survey,” *Comp. Vision Graphics and Image Understanding*, 73(1): 82–97, 1999.

[28] A. Senior, “Real-time articulated human body tracking using silhouette information,” *Proc. Works. Visual Surveillance / PETS*, 2003.

[29] C. Bregler and J. Malik, “Tracking people with twists and exponential maps,” in *Proc. Conf. Comp. Vision Pattern Recog.*, 1998.

[30] H.A. Rowley, S. Baluja, and T. Kanade, “Neural network-based face detection,” *IEEE Trans. Pattern Anal. Machine Intell.*, 20(1): 23–28, 1998.

[31] E. Osuna, R. Freund, and F. Girosi, “Training support vector machines: An application to face detection,” in *Proc. Conf. Computer Vision Pattern Recog.*, pp. 130–136, 1997.

[32] D. Roth, M.-H. Yang, and N. Ahuja, “A SNoW-based face detector,” in *Proc. NIPS*, 2000.

[33] P. Viola and M. Jones, “Robust real time object detection,” in *Proc. IEEE ICCV Work. Statistical and Computational Theories of Vision*, 2001.

[34] H.P. Graf, E. Cosatto, and G. Potamianos, “Robust recognition of faces and facial features with a multi-modal system,” *Proc. Int. Conf. Systems Man Cybern.*, Orlando, FL, pp. 2034–2039, 1997.

[35] T.F. Cootes, G.J. Edwards, and C.J. Taylor, “Active appearance models,” *IEEE Trans. Pattern Anal. Machine Intell.*, 23(6): 681–685, 2001.

[36] A.P. Pentland, B. Moghaddam, and T. Starner, "View-based and modular eigenspaces for face recognition," in *Proc. Conf. Computer Vision Pattern Recog.*, pp. 84–91, 1994.

[37] T.F. Cootes, G.V. Wheeler, K.N. Walker, and C.J. Taylor, "Coupled-view active appearance models," in *Proc. British Machine Vision Conf.*, vol. 1, pp. 52–61, 2000.

[38] S.Z. Li and Z. Zhang, "FloatBoost learning and statistical face detection," *IEEE Trans. Pattern Anal. Machine Intell.*, 26(9): 1112–1123, 2004.

[39] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density," in *Proc. Europ. Conf. Computer Vision*, pp. 343–356, 1996.

[40] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift," in *Proc. Int. Conf. Computer Vision Pattern Recog.*, vol. 2, pp. 142–149, 2000.

[41] H. Tao, H.S. Sawhney, and R. Kumar, "Dynamic layer representation with applications to tracking," in *Proc. Int. Conf. Computer Vision Pattern Recog.*, vol. 2, pp. 134–141, 2000.

[42] M.J. Black and A. Jepson, "Eigentracking: Robust matching and tracking of articulated objects using a view-based representation," *Int. J. Computer Vision*, 26(1): 63–84, 1998.

[43] A.D. Jepson, D.J. Fleet and T.F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, 25(10): 1296–1311, 2003.

[44] R.T. Collins, Y. Liu and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Machine Intell.*, 27(10): 1631–1643, 2005.

[45] B. Han and L. Davis, "On-line density-based appearance modeling for object tracking," in *Proc. Int. Conf. Computer Vision*, Beijing, 2005.

[46] R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[47] O. Lanz, "Approximate Bayesian multibody tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, 28(9): 1436–1449, 2006.

[48] D.N. Zotkin, R. Duraiswami, and L.S. Davis, "Joint audio-visual tracking using particle filters," *EURASIP J. Appl. Signal Processing*, 2002(11): 1154–1164, 2002.

[49] A. Mittal and L. Davis. "M2Tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo," in *Proc. European Conf. Comp. Vision*, pp. 18–36, 2002.

[50] R. Rosales and S. Sclaroff, "3D trajectory recovery for tracking multiple objects and trajectory guided recognition of actions," in *Proc. Int. Conf. Comp. Vision Pattern Recog.*, pp. 117–123, 1999.

[51] R. Collins, A. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, and O. Hasegawa, "A system for video surveillance and monitoring: VSAM final report," *Technical Report* CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University, 2000.

[52] J. Black and T. Ellis, "Multi camera image tracking," in *Proc. IEEE Work. on Performance Evaluation of Tracking and Surveillance*, 2001.

[53] Q. Zhou and J. K. Aggarwal, "Object tracking in an outdoor environment using fusion of feature and cameras," [To appear:] in *Image and Vision Comp.*

[54] R.E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME – J. Basic Engin. (Ser. D)*, 82: 35–45, 1960.

[55] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *Int. J. Computer Vision*, 29(1):5–28, 1998.

[56] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, 50(2): 174–188, 2002.

[57] C. Stauffer and W.E.L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, 22(8): 747–757, 2000.

[58] A. Tyagi, G. Potamianos, J.W. Davis, and S.M. Chu, "Fusion of multiple camera views for kernel-based 3D tracking," [To Appear:] *Proc. IEEE Works. Motion and Video Computing*, Austin, Texas, 2007.

[59] J. Ho, K.-C. Lee, M.-H. Yang, and D. Kriegman, "Visual tracking using learned linear subspaces," in *Proc. Int. Conf. Computer Vision Pattern Recog.*, vol. 1, pp. 782–789, 2004.

[60] P. Hall, D. Marshall, and R. Martin, "Merging and splitting eigenspace models," *IEEE Trans. Pattern Anal. Machine Intell.*, 22(9): 1042–1049, 2000.

[61] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Computer and System Sciences*, 55(1): 119–139, 1997.

[62] K. Tieu and P. Viola, "Boosting image retrieval," *Proc. Conf. Computer Vision Pattern Recog.*, vol. 1, pp. 228–235, 2000.

[63] H. Schneiderman and T. Kanade, "A statistical method for 3D object detection applied to faces and cars," *Proc. Conf. Computer Vision Pattern Recog.*, 2000.

[64] P. Pudil, J. Novovicova, and J. Kittler, "Floating search methods in feature selection," *Pattern Recog. Lett.*, 15:1119–1125, 1994.

[65] J. Connell, A.W. Senior, A. Hampapur, Y.-L. Tian, L. Brown, and S. Pankanti, "Detection and tracking in the IBM PeopleVision system," in *Proc. Int. Conf. Multimedia Expo (ICME)*, 2004.

[66] A. Bobick and J. Davis, "The representation and recognition of action using temporal templates," *IEEE Trans. Pattern Anal. Machine Intell.*, 23(3): 257–267, 2001.

[67] Y. Boykov and D. Huttenlocher, "Adaptive Bayesian recognition in tracking rigid objects," in *Proc. Int. Conf. Comp. Vision Pattern Recog.*, pp. 697–704, 2000.

[68] A. Senior, "Tracking with probabilistic appearance models," in *Proc. Int. Work. on Performance Evaluation of Tracking and Surveillance Systems*, 2002.

[69] A.W. Senior, G. Potamianos, S. Chu, Z. Zhang, and A. Hampapur, "A comparison of multicamera person-tracking algorithms," *Proc. IEEE Int. Work. Visual Surveillance (VS/ECCV)*, Graz, Austria, 2006.

[70] VACE: "Video Analysis and Content Extraction" [Online]. Available: `http://www.nbc.gov/fort_h/vaceiii.html`

[71] K. Bernardin, A. Elbs, and R. Stiefelhagen, "Multiple object tracking performance metrics and evaluation in a smart room environment," in *Proc. Int. Works. Visual Surveillance*, Graz, Austria, 2006.