Tracking the Activity of Participants in a Meeting

Hammadi Nait Charif and Stephen J. McKenna
School of Computing, University of Dundee,
Dundee DD1 4HN, Scotland
Tel. +44 1382 344732
Fax. +44 1382 345509
stephen@computing.dundee.ac.uk

Published in the journal Machine Vision and Applications 17(2):83-93, 2006.
The original publication is available at www.springerlink.com.

Abstract

A vision system suitable for a smart meeting room able to analyse the activities of its occupants is described. Multiple people were tracked using a particle filter in which samples were iteratively re-weighted using an approximate likelihood in each frame. Trackers were automatically initialised and constrained using simple contextual knowledge of the room layout. Person-person occlusion was handled using multiple cameras. The method was evaluated on video sequences of a six person meeting. The tracker was demonstrated to outperform standard sampling importance re-sampling. All meeting participants were successfully tracked and their actions were recognised throughout the meeting scenarios tested.

1 Introduction

The idea of the 'smart' meeting room, able to unobtrusively sense and interpret the rich stream of activity of its occupants, has generated significant recent interest. The reader is referred, for example, to the Multi-Modal Meeting Manager (M4) project (EU IST-2001-34485) and recent workshops with emphasis on evaluation of tracking for smart meeting rooms [11] and analysis of meeting data [2]. The idea is beginning to show promise as a realisable and potentially highly useful technological application. A successful smart meeting room with robust operation will probably require multiple sensing modalities. Desirable capabilities include localisation, identification and tracking of meeting participants as well as recognition of their speech, gestures, actions, emotions and their focus of attention [4, 13, 26, 27, 28]. A meeting room endowed with these perceptual abilities could provide advanced meeting support services, reacting appropriately to users' needs and facilitating effective interaction during a meeting. Another important strand of applications is archiving, indexing and retrieval. We often forget information shared at meetings and whilst written meeting minutes provide retrieval cues for our human memories, it seems clear that audio-visual information can provide a far richer record, provided that such data can be appropriately (automatically) annotated and efficiently retrieved [17].

This paper makes a contribution towards achieving a subset of the above goals by describing a computer vision system that tracks participants in a meeting room and recognises some of their actions. The system was evaluated using the PETS-ICVS [11] smart meeting room video sequences. The aim was to au-

H. Nait Charif was funded by UK EPSRC Grant GR/R27419/01

tomatically track and annotate certain actions of the meeting participants based on video data. A premise of this work was that reliable tracking of the head of each person would yield interesting annotation data in terms of motion trajectories and that these in turn could be used to recognise actions such as standing up, sitting down, entering, exiting and walking to the whiteboard. The system needed to be able to simultaneously track multiple people, perform automatic initialisation, handle person-person occlusion and combine data from two cameras to annotate the activity of all six participants throughout a meeting.

The remainder of this paper is organised as follows. Section 2 briefly reviews some related work on tracking using particle filters. Section 3 describes the Sampling Importance Resampling (SIR) filter and discusses its limitations. Section 4 describes and motivates the modified filter, Iterated Likelihood Weighting (ILW) used here. Section 5 describes the head model and the approximate likelihood measurement based on combined region (color) and boundary (gradient) cues. The mechanisms used for initialisation and occlusion handling are described in Section 6. Empirical results are reported in Section 7 for tracking (using SIR, ICondensation and ILW) and action recognition. Finally, some conclusions are drawn in Section 8.

2 Tracking using Particle Filters

Visual tracking is often formulated from a Bayesian perspective as a problem of estimating some degree of belief in the state $\mathbf{x}_t$ of an object at time step t given a sequence of observations $\mathbf{z}_{1:t}$. Bayesian filtering recursively computes a posterior density that can be written using Bayes rule as:

$$p(\mathbf{x}_{t+1} | \mathbf{z}_{t+1}) \propto p(\mathbf{z}_{t+1} | \mathbf{x}_{t+1}) p(\mathbf{x}_{t+1}) \quad (1)$$

Applying a Markov assumption, the prior density is the posterior density propagated from the previous time step using a dynamic model:

$$p(\mathbf{x}_{t+1}) = \int p(\mathbf{x}_{t+1} | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{z}_t) d\mathbf{x}_t \quad (2)$$

The posterior in (1) cannot be computed analytically unless linear-Gaussian models are adopted, in which case the Kalman filter provides the solution. As is well-known, linear-Gaussian models are unsuitable for tracking in visual clutter. Instead, particle filters are often used to propagate what are often non-Gaussian, multimodal densities over time. A modification to the frequently used Sampling Importance Resampling algorithm was used here to improve the accuracy and consistency of tracking [21].

Isard and Blake suggested particle filtering for visual tracking in the form of Condensation [14]. Particle filtering [12] is now popular for tracking and several authors have suggested alternative sampling schemes. For example Choo and Fleet [6] used a hybrid Monte Carlo filter to sample the posterior for human tracking. Rui and Chen [25] used an unscented Kalman filter to generate importance densities for particle filter-based tracking. Deutscher *et al.* [9, 10] proposed annealed and partitioned particle filtering for human tracking. Isard and MacCormick [16] described a multiple blob tracker with a prediction algorithm based on there being a fixed probability of a new object entering the scene at each time step. Sample positions for any such new object were drawn uniformly over a fixed region of the scene. Thus some particles at each time step could be in regions of state space with low temporal priors as determined by previously tracked objects. Isard and Blake [15] used importance sampling to fuse information from a skin color detector with a contour tracker. In this approach, the effect of the color cue was to ensure that samples were made in image regions of appropriate color which might otherwise not have been sampled sufficiently due to their temporal prior density being low. The auxiliary particle filter of Pitt and Shephard [23] uses a proposal distribution that is a mixture that depends on both the past state and the most recent oservation. Other variations on basic particle filtering have been proposed outside the vision literature (see e.g. [5, 20]). Arulampalam *et al.* [1] provide a useful tutorial.

3 Sampling Importance Resampling

Sampling Importance Resampling (SIR) [12] (Condensation [14]) approximates the posterior density $p(\mathbf{x}_t|\mathbf{z}_t)$ at each time step t by a set of N particles $\{\mathbf{x}_t^n, w_t^n\}_{n=1}^N$ where each particle is a weighted random sample and $\sum_{n=1}^N w_t^n = 1$. The filtered posterior is then

$$p(\mathbf{x}_{t+1}|\mathbf{z}_{t+1}) \propto p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1}) \sum_{n=1}^{N} w_t^n p(\mathbf{x}_{t+1}|\mathbf{x}_t^n) \quad (3)$$

where the prior is now a mixture with N components. The SIR filter involves (i) selecting the n^{th} mixture component with probability w_t^n, (ii) drawing a sample from it, and (iii) assigning the sample a weight proportional to its likelihood. Resampling is used to obtain samples with equal weights in order to facilitate sampling from the mixture in (3). The algorithm is given in Table 1 for completeness. The dynamic (motion) model is encapsulated by the transition density $p(\mathbf{x}_{t+1}|\mathbf{x}_t^n)$. Typically, a sample can be drawn from it by adding random process noise and then applying deterministic dynamics (drift).

In general, sequential importance sampling filters operate by drawing samples from an importance density, $q(\mathbf{x})$, and weighting them using (4) to give a particle representation of the posterior density.

$$w_{t+1}^n \propto w_t^n \frac{p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1}^n)p(\mathbf{x}_{t+1}^n|\mathbf{x}_t^n)}{q(\mathbf{x}_{t+1}^n|\mathbf{x}_t^n, \mathbf{z}_{t+1})} \quad (4)$$

The SIR filter is an example of a sequential importance sampling filter in which the prior is used as the importance density. This is a convenient choice because an unbiased, asymptotically correct estimate of the posterior can be obtained by simply weighting the samples with their likelihood. The resulting algorithm is therefore intuitive and easily implemented. However, the prior is certainly not the optimal choice of importance function since it does not take into account the most recent observation, $\mathbf{z}_{t+1}$. Sampling using SIR is particularly inefficient when the likelihood is in the tails of the prior or if the likelihood is narrow and peaked compared to the prior. Although SIR gives an asymptotically correct estimate of the posterior, its behaviour with finite sample sets is often not good. Expectations computed using SIR have high variance so that different runs of the tracker can lead to very different results. In human tracking, the dynamic models used can often result in poor priors due to unexpected motion. In such cases, SIR will place many samples in the wrong regions of the state space. As a result, very large particle sets can be required in order to achieve acceptable performance. King and Forsyth [18] comment that SIR "will appear to be following tight peaks in the posterior even in the absence of any meaningful measurement".

Table 1: The Sampling Importance Resampling Algorithm

Draw samples $\mathbf{x}_{t+1}^n \quad p(\mathbf{x}_{t+1}|\mathbf{x}_t^n)$
Assign weights $w_{t+1}^n = p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1}^n)$
Normalise weights so that $\sum_{n=1}^N w_{t+1}^n = 1$
Resample with replacement to obtain
samples $\mathbf{x}_{t+1}^n$ with equal weights, $(w_{t+1}^n = 1/N \quad \forall n)$

SIR's use of the prior as the importance function in Equation (4) results in a simplified algorithm but ignores the most recent observation when sampling. An alternative approach named ICondensation [15] generates some of the samples as in SIR and some of the samples using an importance function that depends on the most recent observation but ignores the dynamics, i.e. $q(\mathbf{x}_{t+1}^n|\mathbf{x}_t^n, \mathbf{z}_{t+1}) = q(\mathbf{x}_{t+1}^n|\mathbf{z}_{t+1})$. Isard and Blake [15] demonstrated a hand tracking system using this approach with a contour-based likelihood model and an importance function based on skin color blob detection.

4 Iterated Likelihood Weighting

Great care is usually taken to ensure that an unbiased estimate of the posterior is obtained. The importance sampling step of (4) is a bias-correcting scheme used to obtain such an unbiased estimate. However, approximation error depends not only on the bias but on the variance (the bias-variance dilemma). If the

importance density is reasonably accurate, the correction step may in fact increase the approximation error for all but very large particle sets (see [29] for examples of this phenomenon). In other words, bias is reduced at the cost of higher variance which can lead to a poorer approximation. Furthermore, the prior density is often poor and noisy and it therefore makes little sense to attempt to obtain a computationally expensive, high accuracy approximation to the posterior. This is particularly true in many human tracking applications where inter-frame motion is often poorly modeled by the dynamic model (transition density).

A scheme is used here in which only a subset of the particles at each time step are sampled from the 'posterior'. The remaining particles are used to increase sampling in regions of high likelihood via a simple iterative search using the most recent observation. This is useful when the prior is poor and can prevent tracking failure in the case of unexpected motion, for example. Rather than attempt a (potentially expensive) bias-correction step for those particles used to search high-likelihood regions, they are weighted at each iteration based on their likelihood. The resulting particle set is not, asymptotically, an unbiased representation of the posterior. The algorithm is asymptotically biased. It can be thought of as SIR combined with an iterative application of SIR several times on the same observation. The algorithm, called Iterated Likelihood Weighting (ILW), is given in Table 2. After an intial iteration of SIR, the sample set is split uniformly at random into two sets of equal size. One of these sets is propagated to the next time step unaltered while the samples in the other set are subjected to further iterations of diffusion, likelihood weighting and resampling. Given the broad likelihood responses, this has the effect of migrating half of the particles to regions of high likelihood while the other half are sampled using the prior as the importance function. In a situation where the prior is good, its use as an importance function by half the particles will result in useful samples. However, if the prior is poor, the iterated particle set will still explore regions of high likelihood.

In the special case of a Gaussian transition density, the ILW diffusion step (step 6.(a) in Table 2) amounts to applying these transition dynamics. In general, however, only step 1. of the algorithm applies dynamics.

Table 2: The Iterated Likelihood Weighting Filter. Here $p(\mathbf{x}_{t+1;k+1} | \mathbf{x}^m_{t+1;k})$ is a transition density with expected value $\mathbf{x}^m_{t+1;k}$.

1. Draw N samples $\mathbf{x}^n_{t+1} \quad p(\mathbf{x}_{t+1} | \mathbf{x}^n_t)$
2. Assign weights $w^n_{t+1} = p(\mathbf{z}_{t+1} | \mathbf{x}^n_{t+1})$
3. Normalise weights so that $\sum_{n=1}^{N} w^n_{t+1} = 1$
4. Resample with replacement to obtain samples $\mathbf{x}^n_{t+1}$ with equal weights
5. Split the sample set at random into two sets of size $M = N/2$: $\{\mathbf{x}^m_{t+1;1}\}_{m=1}^{M}$ and $\{\mathbf{x}^m_{t+1;}\}_{m=1}^{M}$
6. For $k = 1 : : : K$
   (a) Draw M samples $\mathbf{x}^m_{t+1;k+1} \quad p(\mathbf{x}_{t+1;k+1} | \mathbf{x}^m_{t+1;k})$
   (b) Assign weights $w^m_{t+1;k+1} = p(\mathbf{z}_{t+1} | \mathbf{x}^m_{t+1;k+1})$
   (c) Normalise weights so that $\sum_{m=1}^{M} w^m_{t+1;k+1} = 1$
   (d) Resample with replacement to obtain M samples $\mathbf{x}^m_{t+1;k+1}$ with equal weights
7. For $m = 1 : : : M$
   $\mathbf{x}^m_{t+1} = \mathbf{x}^m_{t+1;}$
   $\mathbf{x}^{M+m}_{t+1} = \mathbf{x}^m_{t+1;K+1}$

5 Head Model

In order to apply the above filtering scheme to the tracking problem, the state vector, $\mathbf{x}$, and the likelihood model, $p(\mathbf{z} | \mathbf{x})$, must be defined. A well designed likelihood model can significantly improve tracking performance [24].

Head shape is reasonably well approximated as an ellipse in the image irrespective of pose. Rui and Chen [25] used a fixed ellipse and tracked its 2D translation using Canny edges. Nummiaro *et al.* [22] used an ellipse with fixed orientation and a likelihood based only on color. Birchfield [3] used an ellipse constrained to be vertically-oriented and of a fixed eccentricity leaving only three parameters to be estimated. Here eccentricity is allowed to vary with pose and position while orientation is fixed. Therefore, four ellipse parameters were estimated.

The approximate likelihood model used combines intensity gradient information along the head boundary with a color model of the ellipse's interior region. The color-based measurement (x_t^n) is obtained by computing the intersection of a 3-D color histogram of the ellipse's interior and a stored model color histogram. Histograms were formed in RGB space with 8 8 8 bins. The gradient-based measurement (x_t^n) involves searching for maximum gradient magnitude points along short radial search line segments centered on the ellipse boundary. There are 30 such lines, each 5 pixels long. The overall likelihood was formulated heuristically as:

$$p(\mathbf{z}_t|x_t^n) = \frac{(\ (x_t^n)\ (x_t^n))^2}{\sum_{n=1}^{N}(\ (x_t^n)\ (x_t^n))^2} \qquad (5)$$

This has characteristics preferable to the use of boundary cues or region cues alone. The boundary cue alone results in a noisy reponse with many local maxima. The region cue alone results in a response that varies more slowly with translation but which does not decrease appropriately with reduced scale. The combined cue, on the other hand, gives a clear maximum in the correct location and varies in a well-behaved manner as both translation and scale change.

6 Initialisation and Occlusion Handling

Tracker initialisation and occlusion handling made use of scene-specific contextual information as illustrated in Fig. 1. The room layout and the maximum height of a person meant that the heads of people on the far side of the table always appeared between the upper and lower horizontal lines in Fig. 1. Therefore, no particles were ever propagated outside this bounding box.

When people pass in front of the camera on the opposite wall (visible in the upper centre of Fig. 1) they occlude the view of the people on the near side of the table from that opposing camera. The boxes near the centre of Fig. 1 indicate regions in which this occurs. When such an occlusion event is detected, any tracks

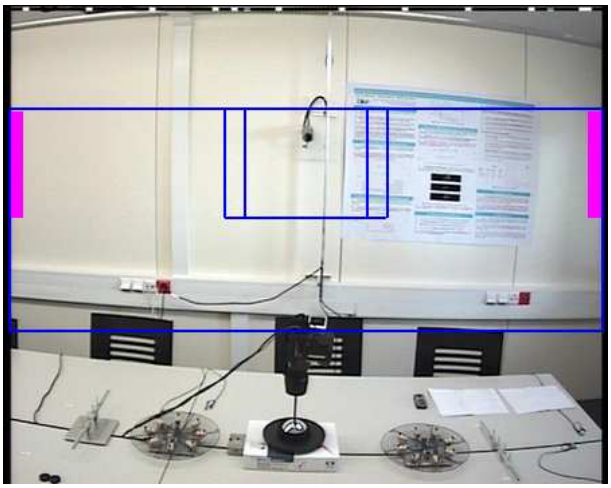

Figure 1: Scene constraints used to perform tracking, initialisation and occlusion handling.

in the corresponding regions of the opposing camera's field of view are suspended until the occlusion is over. Provided that the occluded people do not move too much while occluded, their trackers will recover and continue to track them.

Initialisation was performed in expected entry and exit regions indicated in Fig. 1 by filled rectangles. A background subtraction algorithm was applied in each frame within these regions. Whenever significant change was detected, an initial particle set of head ellipses was instantiated centred within the region and a tracker was initialised. Color histograms were learned from a single frontal head view taken from elsewhere in the PETS-ICVS data set. No histogram adaptation was needed. When a tracker's estimated head ellipse left the field of view in the direction of the whiteboard, that tracker waited for the background subtraction routine to signal re-entry. When a tracker's estimated head ellipse left the view in the direction of the exit, the tracker was terminated.

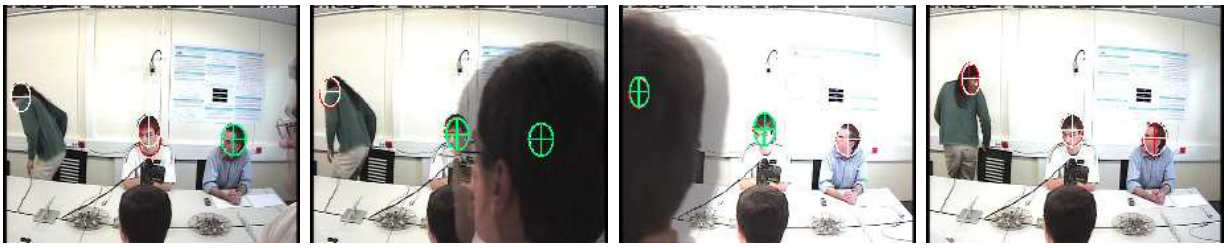

Figure 2: Occlusion handling. Frames 17125, 17130, 17135 and 17140 of Scenario C viewed from Camera 1. Suspended tracks are shown as green ellipses. (The halo effect around the occluding head is due to interlacing.)

7 Evaluation

The tracking methods were implemented using a Gaussian transition density with diagonal covariance matrix. Specifically, the variance parameters were $^2 = 11^2$ pixels for the ellipse centre parameters and $^2 = 2^2$ pixels for the ellipse major and minor semi-axis parameters. In common with similar previous work on particle filter tracking, a person can be tracked comfortably in real-time (25H z) on a standard PC. Computational cost scales linearly with the number of people to be tracked. The current C++ implementation cannot simultaneously track all the meeting participants in real-time. However, this goal has not been pursued. The method is amenable to parallel implementation.

7.1 The Meeting Room and Video Data Sets

The method was evaluated using sequences provided by the consortium of Project FGnet (IST-2000-26434) for the PETS-ICVS Workshop [11]. Image size was resampled to 450 360 pixels. The meeting room had a table with seating for six participants, three seats on each side of the table. Video was provided from two wall-mounted cameras on opposite sides of the meeting room. Scenario C ("Going to the white board") in particular was used to illustrate performance (Figs. 2– 7). This scenario began with each of six meeting participants in turn entering and then sitting down. Subsequently, each in turn stood up, walked to the whiteboard, wrote something and then returned to his seat, twice. Finally, each person in turn exited the room. This scenario and a second scenario (B: "Performing Face and Hand Gestures") were used to evaluate the performance of the system in terms of its ability to track each of the people throughout entire meeting sequences. Furthermore, its ability to recognise the actions of entering, exiting, sitting down, getting up, and going to the whiteboard was evaluated.

7.2 Head Tracking Results

The method of initialisation, reinitialisation and termination was 100% successful on both Scenarios B and C. There were no false initialisations, people were always tracked after returning from the whiteboard and trackers were terminated only when people left the room.

The occlusion handling mechanism was successful in all the sequences tested here. Fig. 2 shows an example in which each of the trackers for the three people on the far side of the room were suspended and recovered in turn due to a person near the camera moving left to right across its field of view.

All the meeting participants were successfully tracked throughout both the Scenario B and C sequences using ILW. Figs. 3 and 4 show a selection of frames from each of the two camera views for Scenario C. For each person, a red ellipse is used to indicate the mean estimated from the particle set and a white ellipse to indicate the mean of the 10 most heavily

Figure 3: Frames 10950, 11270, 13630, 13720, 14280, 14840, 17200 and 19100 of Scenario C from Camera 1 in which persons (4), (5) and (6) are tracked using ILW entering the room, sitting down, going to the white-board (to the right of the field of view) returning to their seats and finally exiting the room (to the left of the field of view).

Figure 4: Frames 10500, 11065, 11940, 12810, 13615, 15790, 16500 and 19200 of Scenario C from Camera 2 in which persons (1), (2) and (3) are tracked using ILW entering the room, sitting down, going to the white-board (to the left of the field of view), returning to their seats and finally exiting (to the right of the field of view).

weighted particles for that frame. In these examples, these two estimates were very similar. The mean of the 10 strongest particles gave a temporally smooth estimate of the location of the strongest mode of the distribution. Fig. 5 shows the trajectories of the centres of each person's head for the entire sequence.

Standard SIR filtering performed relatively poorly. Fig. 6 shows example frames from a typical run

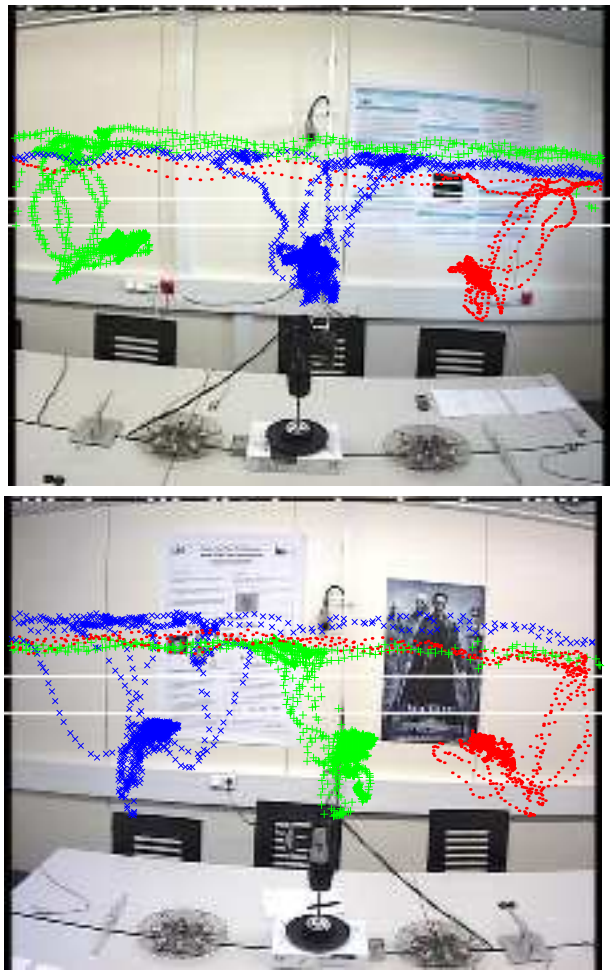

Figure 5: The estimated trajectories for the head centres for the entire duration of Scenario C. The positions are shown for each frame using a different color for each person, overlaid on images of the empty meeting room. The upper image shows the view from Camera 1 and the lower image the view from Camera 2. The horizontal white lines were used for action recognition.

Figure 6: Frames 10400, 10500, 10550, 11070, 11090 and 11300 of Scenario C seen by Camera 2. Here the SIR tracker loses person 1 and later tracks person 3 with two ellipses simultaneously.

with Camera 2. Trackers lost lock, became attached to background clutter and subsequently tracked the wrong person. The performance of SIR was better with Camera 1 because there was less background clutter. However, tracking failures in the Camera 2 view confused the occlusion management algorithm which in turn led to the trackers in Camera 1 failing.

Although the above runs were typical for these sequences, isolated runs of particle filters are not sufficient to evaluate performance due to the variance of the filters. Likelihood computation is the main computational expense during tracking and the different filters require different numbers of likelihood evaluations per frame. In order to obtain a fair empirical comparison, the number of particles used with each filter was chosen so that the number of likelihood evaluations per frame was equal, i.e. 200 particles for the SIR, and 50 for ILW, respectively. ILW used 6 iterations. Likelihood evaluations per frame were thus the same for each method. Both filters were run with the same transition density (a Gaussian centered on

Figure 7: Head estimates at frame 10520 of Scenario C after 50 different runs of the tracking algorithm starting from the first frame of the sequence. The upper image shows the results using SIR and the lower image the results using ILW. Each ellipse is the mean of the ten strongest particles.

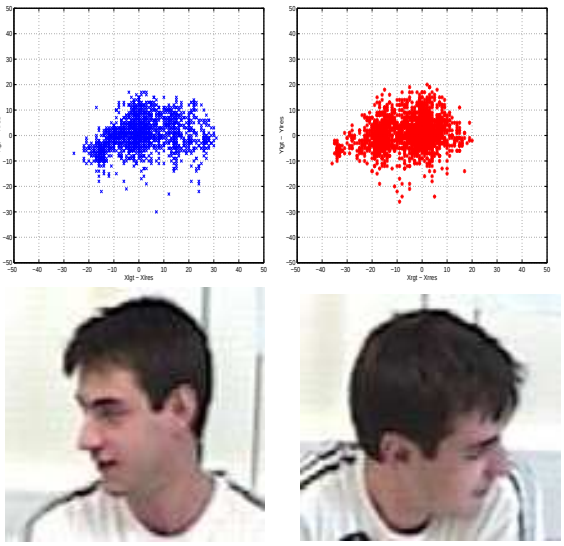

Figure 8: Top: Distributions of the displacement errors for the left and right eyes. Ground-truth is at the origin in the centre of the plots. Bottom: Example images at the same scale as the plots illustrating the extent of head rotation.

the previous sample) and the same noise parameters. Fig. 7 compares multiple runs of the two filters on the same sequence. The SIR filter failed in the majority of runs. The SIR trackers for person 1 and person 2 maintained lock in only 30% and 38% of runs respectively. At least one of the SIR trackers lost lock in 84% of the runs. In contrast, ILW lost lock in only 2% of cases. One of the ILW trackers always maintained lock.

Ground-truth data for the eye positions were available for a section of Scenario B in which the participants were seated. While the system was never intended to accurately estimate eye positions, these data were nevertheless used to provide an indication of the accuracy and consistency of head tracking under an upright, frontal face view assumption. Performance was quantitatively evaluated using these ground-truth data under the simplistic assumption that the face was always in a frontal, upright view. The eye positions can be estimated under this assumption relative to the head ellipse. This method will clearly become innaccurate when the assumption is violated by head rotation. The distribution of displacement errors is shown in Fig. 8. The mean displacement in the y-direction was 0.5 pixels with a standard deviation of $= 6.3$. This variance was due to both slight head tilt violating the frontal view assumption and small head tracking errors. Given that a head appears approximately 80 pixels in height, the error is relatively small and indicates accurate head tracking. Errors in the x-direction were of course larger due to the severe violation of the frontal view assumption when people turned their heads from side to side. The mean displacements for the left eye and

right eye were 1.7 pixels and 5.6 pixels with standard deviations of 11.5 and 10.9 respectively. This can be regarded as a baseline performance against which to compare eye position estimators. The estimates were always within the true head region.

7.3 Comparison with ICondensation

A method based on ICondensation [15] in which the most recent observation is used to generate an importance function, was also implemented for head tracking. In particular, an importance function was generated from the current frame as a 10-component mixture of 2 D Gaussians. This mixture was fitted using Expectation-Maximization to a pixel color likelihood computed using the head color histogram. Similarly to Isard and Blake [15], a mixed sampling scheme was adopted. At each frame, half the samples were generated using standard factored sampling (as in SIR) and half using color importance sampling. In the latter case, the ellipse state was first sampled using standard factored sampling and the translation parameters were subsequently replaced by sampling from the color importance function.

Figure 11: Head estimates at frame 10571 of Scenario C after 20 different runs of ICondensation starting from the first frame of the sequence. For each run, a red ellipse denotes the mean for the tracker initialised on the leftmost person and the white ellipse the mean for the tracker initialised on the other person.

Figs. 9 and 10 illustrate example results computed using ICondensation. In both Figures, a tracker initialised on one person jumps to another person when the color cue becomes temporarily unreliable due to unusual head poses. Fig. 11 shows results obtained over multiple runs using ICondensation with 400 particles. Although two trackers were initialised, one per person, an extreme head pose resulted in both trackers locking onto to the same person in every run. This behaviour can be explained by considering the global nature of the color importance function. When the color cue for a head becomes temporarily unreliable, sampling from the color importance function results in particles being placed on other head colored regions in the scene. In the presence of other similar objects (heads), the dynamic prior is insufficient to constrain the tracker and it jumps to another object. Whilst the global nature of the color importance function can enable robust tracking of large motions, the presence of other similar objects in the scene makes it liable to lose lock on the intended target.

7.4 Action Recognition

Given the reliable head tracking just described, recognition of several actions in the meeting Scenario C became straightforward. These actions were entering, exiting, going to the whiteboard, getting up and sitting down. The first three can be recognised by detecting where trackers initialise and terminate. Sitting down and getting up can be recognised by detecting when the head centre crosses the horizontal lines shown in Fig. 5. In particular, a person is classified as sitting if their head is below the lower line and as standing if their head is above the upper line. When between the two lines, they are classified as transitioning between sitting and standing, i.e. "sitting down" or "getting up". Table 3 gives a detailed action annotation obtained. This is given here to allow other researchers to compare their results on this public domain data set. This temporal segmentation into actions was qualitatively correct throughout both Scenarios B and C. No actions were falsely detected or missed. The order of events was recovered correctly.

This action recognition method obviously relies

Figure 9: An ICondensation tracker locking on to the wrong person. An unusual head pose resulted in a poor importance function causing the tracker to jump to another person rather than track through this temporary change in color distribution.

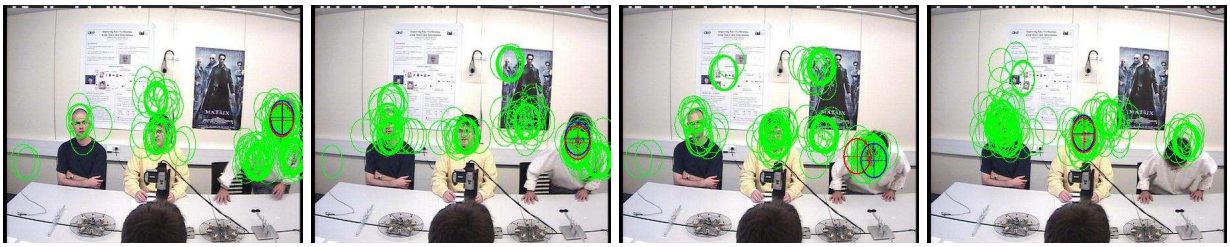

Figure 10: Four images from a run of a single ICondensation tracker showing the (unweighted) particle sets propagated. This tracker was initialised for the person on the right side of the image. The importance function results in large numbers of samples placed on the other people and on background color clutter. A temporary unusual head pose results in the tracker jumping to another person who is tracked from then on.

heavily on scene-specific constraints. Since only a few examples of these actions occurred in the data provided it was not possible to properly evaluate the method's ability to generalise. Methods based on learning more complex models of action become feasible only with larger data sets of example actions. The approach adopted here was to fit a simple method to the available data. Recognition of these actions was made relatively simple by virtue of the success of the head tracker.

8 Conclusions

All meeting participants were successfully tracked throughout long image sequences with automatic initialisation and termination of tracking. They were tracked through occlusion using views from two different cameras. Given some simple scene-specific constraints, the tracking results enabled the actions of entering, exiting, going to the whiteboard, sitting down and getting up to be recognised. All such actions were detected without false detections.

Two trackers based on SIR and ILW were compared. The experiments show that the variance of SIR can be high while the approximation accuracy is often poor. The ILW tracker yielded better accuracy and lower variance. There are several other particle filtering schemes that should also give better performance than standard SIR. For example, the auxiliary particle filter uses the most recent observation when computing an unbiased estimate of the posterior [23]. In a previous overhead tracking experiment performed by the authors it was, however, outperformed by ILW [21].

It should be noted that there exist tracking methods not based on particle filtering, such as kernel-based tracking as proposed by Comaniciu *et al.* [7, 8], that are likely to perform competitively in this ap-

Table 3: The temporal segmentation obtained of Scenario C into activities.

Activity	Person 1	Person 2	Person 3	Person 4	Person 5	Person 6
Enters room	10306	10487	11061	11219	10676	10852
Walking	10306 - 10408	10488 - 10559	11061 - 11061	11219 - 11303	10677 - 10757	10852
Sitting down	10409 - 10412	10560 - 10564	11062 - 11066	11304 - 11313	10758 - 10765	10929 - 10934
Sitting	10413 - 11905	10565 - 12424	11067 - 12937	11314 - 14821	10766 - 14264	10935 - 13676
Getting up	11906 - 11910	12425 - 2431	12938 - 12946	14822 - 14827	14265 - 14269	13677 - 13680
Walking to whiteboard	11911 - 11979	12432 - 12496	12946 - 13040	14828 - 14850	14270 - 14325	13681 - 13771
At whiteboard	11980 - 12278	12497 - 12790	13041 - 13572	14851 - 15160	14326 - 14684	13772 - 14070
Walking from whiteboard	12279 - 12356	12791 - 12852	13573 - 13651	15161 - 15168	14685 - 14764	14071 - 14183
Sitting down	12357 - 12360	12853 - 12857	13652 - 13658	15189 - 15193	14765 - 15768	14184 - 14190
Sitting	12361 - 15435	12858 - 15890	13659 - 16465	15194 - 18459	15769 - 17733	14191 - 17115
Getting up	15435 - 15438	15891 - 15896	16466 - 16470	18460 - 18469	17734 - 17737	17116 - 17122
Walking to whiteboard	15438 - 15450	15897 - 15986	16471 - 16560	18470 - 18486	17738 - 17824	17123 - 17279
At white board	15450 - 15800	15987 - 16326	16561 - 17100	18487 - 18856	17825 - 18262	17280 - 17514
Walking from whiteboard	15801 - 15827	16327 - 16399	17101 - 17189	18857 - 18866	18263 - 18350	17515 - 17518
Sitting down	15828 - 15833	16400 - 16405	17190 - 17196	18896 - 18904	18351 - 18357	17643 - 17648
Sitting	15834 - 20001	16406 - 19170	17197 - 19827	18905 - 19597	18358 - 19347	17522 - 17642
Getting up	20002 - 20006	19171 - 19177	19828 - 19834	19598 - 19608	19348 - 19360	19005 - 19009
Walking from room	20007 - 20101	19178 - 19269	19835 - 19851	19608 - 19704	19361 - 19455	19010 - 19107
Leaves room	20102	19270	19852	19705	19456	19108

plication. It would be interesting to perform direct empirical comparisons with such methods in future work.

In conclusion, the following points can be made regarding performance evaluation of tracking and surveillance systems such as those used here.

It is not sufficient to compare performance based on single runs, even on long sequences. Instead, multiple runs should be used so that the variation due to the stochastic nature of the algorithms can be analysed.

When comparing different algorithms, free parameters should be adjusted so that computational expense per frame is comparable. In the case of comparing SIR and ILW, this was achieved by varying the size of the particle set so that the number of likelihood evaluations was the same.

When evaluating a tracker, its role in the overall system needs to be considered. In many cases, frame-by-frame accuracy is not as important as ensuring that a tracker does not lose lock altogether. The system presented here did not lose lock at all throughout the sequences tested. The role in the overall system also determines to some extent what ground-truth data are appropriate.

Good tracking performance simplifies action recognition. Simple rules based on head location were sufficient to classify actions for data used here.

In the future, the method proposed here should be evaluated in meeting room environments other than that of the PETS-ICVS data sets. Although the occlusion handling and initialisation methods described worked well for these scenarios, a system applicable to a wide range of meeting room configurations would doubtless require extensions to be made. For example, in the PETS-ICVS sequences, heads being tracked on the far side of the table never occlude one another. A solution to the problem of how to track through such occlusions could make use of clothing color to help disambiguate such situations [19].

References

[1] Arulampalam, S., Maskell, S.R., Gordon, N.J., Clapp, T.: A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Transactions on Signal Processing **50**(2), 174–188 (2002)

[2] Bengio, S., Bourlard, H. (eds.): First International Workshop on Machine Learning for Multimodal Interaction (MLMI), *Lecture Notes in Computer Science*, vol. 3361. Springer, Martigny, Switzerland (2004)

[3] Birchfield, S.: Elliptical head tracking using intensity gradients and color histograms. In: IEEE

Conference on Computer Vision and Pattern Recognition. Santa Barbara, CA (1998)

[4] Busso, C., Hernanz, S., Chu, C.W., Kwon, S., Lee, S., Georgiou, P.G., Cohen, I., Narayanan, S.: Smart room: Participant and speaker localization and identification. In: International Conference on Acoustics, Speech, and Signal Processing (ICASSP). Philadelphia (2005)

[5] Carpenter, J., Clifford, P., Fearnhead, P.: An improved particle filter for non-linear problems. IEE Proceedings on Radar and Sonar Navigation **146**, 2–7 (1999)

[6] Choo, K., Fleet, D.J.: People tracking using hybrid Monte Carlo filtering. In: IEEE International Conference on Computer Vision, pp. 321–328. Vancouver (2001)

[7] Comaniciu, D., Ramesh V. amd Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 142–149. South Carolina (2000)

[8] Comaniciu, D., Ramesh V. amd Meer, P.: Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(5), 564–577 (2003)

[9] Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 126–133. South Carolina, USA (2000)

[10] Deutscher, J., Davison, A., Reid, I.: Automatic partitioning of high dimensional search spaces associated with articulated body motion capture. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 669–676. Hawaii (2001)

[11] Ferryman, J. (ed.): Fourth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS-ICVS). Graz, Austria (2003)

[12] Gordon, N., Salmond, D., Smith, A.F.M.: Novel approach to nonlinear and non-Gaussian Bayesian state estimation. IEE Proceedings-F **140**, 107–113 (1993)

[13] Huang, K.S., Trivedi, M.M.: Video arrays for real-time tracking of persons, head and face in an intelligent room. IAPR Workshop on Machine Vision Applications **14**(2), 103–111 (2003)

[14] Isard, M., Blake, A.: Contour tracking by stochastic propagation of conditional density. In: European Conference on Computer Vision, vol. 1, pp. 343–356 (1996)

[15] Isard, M., Blake, A.: Icondensation: Unifying low-level and high-level tracking in a stochastic framework. In: European Conference on Computer Vision, vol. I, pp. 893–908. Freiburg, Germany (1998)

[16] Isard, M., MacCormick, J.: BraMBLe: A Bayesian multiple-blob tracker. In: IEEE International Conference on Computer Vision, vol. 2, pp. 34–41. Vancouver (2001)

[17] Jaimes, A., Omura, K., Nagamine, T., Hirata, K.: Memory cues for meeting video retrieval. In: First ACM workshop on Continuous archival and retrieval of personal experiences (CARPE), pp. 74–85. New York (2004)

[18] King, O., Forsyth, D.A.: How does Condensation behave with a finite number of samples? In: European Conference on Computer Vision, pp. 695–709 (2000)

[19] McKenna, S.J., Jabri, S., Duric, Z., Rosenfeld, A., Wechsler, H.: Tracking groups of people. Computer Vision and Image Understanding **80**, 42–56 (2000)

[20] Musso, C., Oudjane, N., LeGland, F.: Improving regularised particle filters. In: A. Doucet, J.F.G. de Freitas, N.J. Gordon (eds.) Sequential Monte Carlo Methods in Practice. Springer-Verlag, New York (2001)

[21] Nait-Charif, H., McKenna, S.J.: Tracking poorly modelled motion using particle filters with iterated likelihood weighting. In: Asian Conference on Computer Vision, pp. 156–161. Jeju Island, Korea (2004)

[22] Nummiaro, K., Koller-Meier, E., Van Gool, L.: A color-based particle filter. In: A. Pece (ed.) First International Workshop on Generative-Model-Based Vision, vol. 2002/01, pp. 53–60 (2002)

[23] Pitt, M., Shephard, N.: Filtering via simulation: Auxiliary particle filters. Journal of the American Statistical Association **94**(446), 590–599 (1999)

[24] Roberts, T., McKenna, S.J., Ricketts, I.W.: Adaptive learning of statistical appearance models for 3D human tracking. In: British Machine Vision Conference, pp. 333–342. Cardiff (2002)

[25] Rui, Y., Chen, Y.: Better proposal distributions: object tracking using unscented particle filter. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 786–793. Hawaii (2001)

[26] Waibel, A., Schultz, T., Bett, M., Denecke, M., Malkin, R., Rogina, I., Stiefelhagen, R., Yang, J.: SMaRT: The smart meeting room task at ISL. In: International Conference on Acoustics, Speech, and Signal Processing (ICASSP), vol. IV, pp. 752–755 (2003)

[27] Wallhoff, F., Zobl, M., Rigoll, G., Potucek, I.: Face tracking in meeting room scenarios using omnidirectional views. In: International Conference on Pattern Recognition (ICPR), pp. IV:933–936. Cambridge, UK (2004)

[28] Zhang, D., Gatica-Perez, D., Bengio, S., McOwan, I., Lathoud, G.: Modeling individual and group actions in meetings: a two-layer hmm framework. In: IEEE Conference on Computer Vision and Pattern Recognition: Workshop on Event Mining in Video (CVPR-EVENT). Washington DC (2004)

[29] Zlochin, M., Baram, Y.: The bias-variance dilemma of the Monte Carlo method. In: G. Dorffner, H. Bischof, K. Hornik (eds.) Artificial Neural Networks: ICANN. Springer Verlag (2001)