Resolving Visual Uncertainty and Occlusion through Probabilistic Reasoning

Jamie Sherrah and Shaogang Gong [*]
Department of Computer Science, Queen Mary and Westfield College
London, E1 4NS, UK
jamie|sgg@dcs.qmw.ac.uk

Abstract

Tracking interacting human body parts from a single two-dimensional view is difficult due to occlusion, ambiguity and spatio-temporal discontinuities. We present a Bayesian network method for this task. The method is not reliant upon spatio-temporal continuity, but exploits it when present. Our inference-based tracking model is compared with a CONDENSATION model augmented with a probabilistic exclusion mechanism. We show that the Bayesian network has the advantages of fully modelling the state space, explicitly representing domain knowledge, and handling complex interactions between variables in a globally consistent and computationally effective manner.

1 Introduction

When tracking multiple overlapping objects under real-world conditions, ambiguities arise due to distracting noise, mis-matching of the tracked objects, and the possibility of occlusion. If the objects are part of the same articulated entity, such as the human body, domain knowledge can be used to resolve some of these ambiguities. A common and robust approach for real-time tracking is to combine multiple visual cues [9, 10]. In the domain of cues such as skin colour and motion, body parts such as hands can become virtually indistinguishable. Therefore joint tracking of the body parts must be performed with an exclusion principle on observations [7, 5]. Often body motion may appear discontinuous since the hands can move quickly and seemingly erratically, therefore methods such as Kalman filtering that are strongly reliant upon well-defined dynamics and temporal continuity are generally inadequate. However, a wide range of domain knowledge is typically available to reduce reliance on spatio-temporal consistency.

In order to deal with these problems of occlusion, ambiguity and discontinuous motion, a framework is required for representing domain knowledge and reasoning about object associations over time without sole reliance on temporal continuity. *Bayesian Networks* [6, 2] provide such a framework, and enable the full set of possible hypotheses to be simultaneously considered in a consistent and probabilistic manner. Using Bayesian Networks, the semantics of occlusion can be represented explicitly. In this work we present a Bayesian Network approach to tracking a person's head and two hands in near frontal views. The advantages of our method are highlighted by comparing it with a CONDENSATION-based tracker that uses an observation exclusion principle.

[*] The authors wish to thank Eng-Jon Ong for generating the hand orientation histograms.

BMVC 2000 doi:10.5244/C.14.26

2 The Nature of Tracking Interacting Body Parts

Tracking the head and potentially overlapping hands of a single person is an ideal example of tracking under ambiguity and occlusion. Given a single near-frontal camera view of the subject, the hands can occlude each other and the face. Using simple visual cues such as motion and skin colour, the two hands are often indistinguishable. Here we take a view-based approach in which motion (*ie:* frame differencing), skin colour and coarse intensity-based orientation measures are extracted from a near-frontal view of a subject. More details can be found in [8]. Under these circumstances, the head skin cluster can be tracked reliably using the mean shift algorithm [1]. A connected components algorithm is then applied to a sub-sampled image to obtain a list of skin-coloured pixel clusters in the image. Tracking hands is subsequently performed at a skin cluster level, and is treated as a temporal association problem. The task requires resolution of the following two questions: (1) which of the skin coloured clusters correspond to hands? and (2) which of the clusters corresponds to the left and right hand respectively?

We simplify the first question by assuming that only the two largest skin clusters other than the head can potentially be hands. Note that one body part can only correspond to one cluster, but a cluster may correspond to one, two or three of the body parts. By considering components from the whole image, tracking is performed in absolute terms so that a hand cannot be ruled out of consideration even under considerable discontinuous motion. A conventional tracker that models the dynamics of the hands would be unable to distinguish between the two hands in general, especially if the assumption of spatio-temporal continuity were frequently violated. We suggest, rather, that the cluster identities of the hands can often be inferred through a process of deduction. Given uncertain and incomplete information, this deduction should be probabilistic rather than rule-based. In the next section, Bayesian Networks are introduced as a framework for representing and using domain knowledge to perform probabilistic inference.

3 Bayesian Networks

A *Bayesian Network* (BN) is a graphical representation of a probability distribution of a set of random variables. Given a set of N variables $\mathbf{X} = X_1, \ldots, X_N$, the joint probability distribution $P(\mathbf{X})$ can be factored in any number of ways using Bayes' rule. A minimal factorisation exploits independencies between variables to exhaustively specify the joint distribution via a sparse set of conditional probabilities. A BN is a directed acyclic graph in which each variable X_i is represented by one node, and directed edges between nodes represent conditional dependencies. Since a dependence is not unique, the connotation of a causal relationship is often attached to each edge, such as "X is the cause of Y", the direction of the edge being associated with the most intuitive relationship between the variables. A BN represents the factorisation:

$$P(\mathbf{X}) = \prod_{i=1}^{N} P(X_i|\text{pa}(i)) \qquad (1)$$

where pa(i) is the set of parent nodes of node i.

Although algorithms exist for automatically structuring a network from training data, BNs are usually constructed by hand. For many applications, this should be seen as an advantage rather than a drawback. Since BNs provide a rich and principled framework for embedding domain knowledge, a user would often prefer to specify the network structure and the conditional probabilities associated with the graph edges, $P(X_i|\mathrm{pa}(i))$. In the case that a node has no parents, conditional probabilities degenerate to priors. The conditional probabilities may be learned from a training set of data, either through statistical sampling in the case of complete data, or using the Expectation-Maximisation algorithm when some variables are unobservable [3].

Given the network structure and parameters, a BN can be used for a variety of tasks, including inference, prediction and marginalisation. To perform inference, the user observes a subset **e** of the N variables, referred to as *evidence*. After incorporating this evidence into the network, the distribution represented is $P(\mathbf{X}|\mathbf{e})$, that is the distribution of all variables given the available evidence. Note that not all variables need to be observed for inference to take place. Given the distribution $P(\mathbf{X}|\mathbf{e})$, marginalisation yields the distribution of each variable given the evidence, $P(X_i|\mathbf{e})$. More practically, one can obtain the most likely joint configuration of the variables given the evidence:

$$\mathbf{x}^* = \overset{\text{argmax}}{\mathbf{x}} \; P(\mathbf{x}|\mathbf{e}) \tag{2}$$

One may wonder how inference across a joint distribution can be tractable for a large number of variables. The answer is in the network structure, which encodes independencies between variables. By exploiting this structure, an accurate and globally consistent representation of $P(\mathbf{X})$ can be obtained through localised computation and message passing [6].

In the case that the network is a poly-tree, inference can be performed using a relatively simple message-passing algorithm [6]. If the network contains undirected cycles, this inference algorithm becomes intractable because the messages can cycle forever. The network must first undergo a series of transformations to obtain a *join tree* in which inference can be performed using message passing [2]. First the graph is moralised by adding an edge between all pairs of parents where none already exists, and then making all edges undirected. The moralised graph is then triangulated by adding fill-in edges until no cycles with more than four edges exist. Maximum cardinality search is then used to turn the triangulated graph into fully-connected groups of nodes called *cliques*. The triangulation of the moralised graph is not unique in general, and finding the triangulation with the smallest cliques is NP-hard. However, the graph transformation process only needs to be performed once off-line. In practice, heuristic greedy algorithms create satisfactorily economical triangulations [2]. Each clique of the triangulated graph corresponds to a node of the join tree, and each edge in the join tree contains a *separator* set of variables:

$$S = C_i \cap C_j \tag{3}$$

where C_i and C_j are adjacent cliques in the join tree. To perform inference, a *potential* function is maintained for each clique, $\{\phi_C, C \in \mathcal{C}\}$, and separator $\{\phi_S, S \in \mathcal{S}\}$. These potentials are maintained through a series of marginalisation operations to jointly represent the global distribution:

$$P(\mathbf{x}) = \frac{\prod_{C \in \mathcal{C}} \phi_C(\mathbf{x}_C)}{\prod_{S \in \mathcal{S}} \phi_S(\mathbf{x}_S)} \tag{4}$$

After initialisation or entry of observed evidence, a two-pass flow propagation algorithm ensures a valid representation of $P(\mathbf{X})$ over the potentials. The space and computational

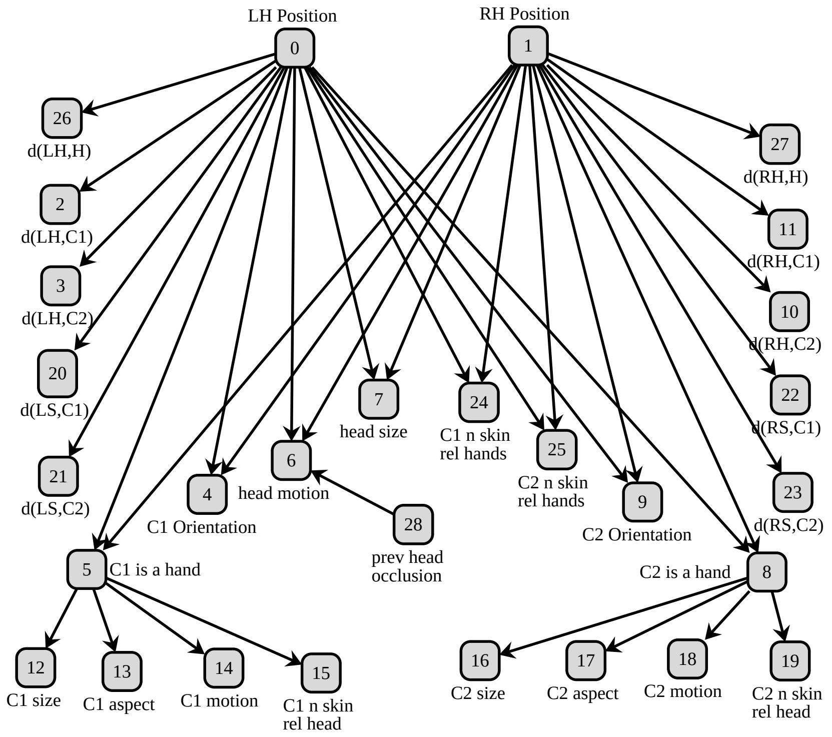


Figure 1: The Bayesian Network for tracking interacting hands. Each node is labelled with an index i and represents a discrete random variable X_i. Text labels briefly describe the variables. Symbols are: LH (left hand), RH (right hand), H (head), C_1 (skin cluster 1), C_2 (skin cluster 2), LS (left shoulder), RS (right shoulder), rel ("relative to").

requirements of the method are proportional to the state space sizes of these cliques. After flow propagation, marginalisation within cliques can be performed to yield $P(X_i)$ for each variable, or a maximisation variation of the propagation algorithm can be used to obtain the most likely configuration. A particular instance of a BN is now described for inferring the head and interacting hand positions of a person.

4 A Network for Tracking Interacting Body Parts

A Bayesian network used for inferring the positions of a subject's hands given the head position is shown in Figure 1. The network consists of 29 discrete variables, all of which are observations except for the abstract quantities X_0, X_1, X_5 and X_8. The total state space size of the set of variables $\mathbf{X} = \{X_0, \ldots, X_{28}\}$ is 9.521245×10^{12}, which is the number of probabilities required to explicitly represent $P(\mathbf{X})$. However, to populate the conditional and prior probability tables of the network required specification of only 456 probabilities, yet any query on the full joint distribution can still be made.

At each frame, the two largest skin clusters that are not the head cluster are identified as C_1 and C_2. The variables to be inferred are X_0 and X_1, which can take on values C_1 (skin cluster 1), C_2 (skin cluster 2), and *Head*. All other variables except X_5 and X_8 are observations made at each frame by discretising continuous values. Note that it is a characteristic of Bayesian networks that inference can still be performed when data are missing. Therefore if there are less than two non-head skin clusters, the variables associated with the unobserved clusters C_i are left uninstantiated and "C_i is a hand" is instantiated to "false". In this way, the dynamically-variable number of skin clusters can be handled without modifying the network structure.

All of the conditional probabilities were specified by hand using common-sense constraints. It is through these probabilities that high-level constraints can be combined in a single framework. For example, the conditional probability table for X_6 (head motion) was used to encode constraints concerning motion, such as:

1. If the head was previously unoccluded by the hands, but now is, then motion must be present in the head region.
2. If at least one hand occluded the head previously and now neither hand occludes the head, then motion must be present in the head region.
3. If the head was previously unoccluded and still is unoccluded, there may still be head motion due to the subject moving his/her head.
4. If only the left hand occluded the head previously, but now only the right hand occludes the head, there must be motion in the head region.

The conditional probabilities are also used to encode boolean rules. For example, the relationship between X_5, "C_1 is a hand", and its parents X_0 and X_1 is:

$$P(X_5|X_0, X_1) = \begin{cases} 1 & \text{if } X_0 = C_1 \text{ or } X_1 = C_1, \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Other constraints exploited are the size (X_{12}, X_{15}, X_{16}, X_{19}), shape (X_{13}, X_{17}) and motion (X_{14}, X_{18}) of the clusters to determine whether they are hands, the distances of the clusters from the shoulders ($X_{20}, X_{21}, X_{22}, X_{23}$) and the previous left- and right-hand positions (X_2, X_3, X_{10}, X_{11}) to determine hand-to-cluster correspondence, the change in head cluster size (X_7) and distance from the head to previous left- and right-hand positions (X_{26}, X_{27}) to determine whether head occlusion has occurred, the appearance-based cluster orientation (X_4, X_9) to determine correct left-right hand associations, and the number of cluster skin pixels relative to the average (X_{24}, X_{25}) to indicate hand occlusion. During inference, we use a maximisation variation of the sum-flow algorithm to establish the most likely configuration of X_0 and X_1 [2].

At the current stage of our work, the BN is a static network in that it performs inference at one time instant only. Nevertheless, the network is coupled indirectly over time through the specification of prior probabilities for X_0 and X_1, the previously-inferred head occlusion state X_{28}, and the distances of the current clusters from the previous hand positions $X_2, X_3, X_{10}, X_{11}, X_{26}$ and X_{27}. Hence tracking is loosely but not solely reliant upon spatio-temporal continuity.

5 Characteristics of Inference-Based Tracking

To highlight the strengths of an inference-based approach to tracking under ambiguity, occlusion and discontinuous motion, we compared our method with a CONDENSATION-based approach that also incorporates an observational exclusion principle, similar to the approach taken in [5]. We refer to the CONDENSATION-based model as the *temporal exclusion* (TE) tracker. The TE tracker state consists of the position and size of a left-hand box and a right-hand box, and a discrete state specifying whether the left hand is on top (closer to the camera than the right hand), or vice versa. It is assumed that both hands are always closer to the camera than the head. As in our approach, the head is tracked using an independent mean-shift tracker. The observation likelihood function $p(\mathbf{z}_t|\mathbf{x}_t^i)$ combines skin colour, motion and orientation cues to validate the location and identity of the hands:

$$p(\mathbf{z}_t|\mathbf{x}_t^i) = (1 + p_o(\mathbf{z}_t|\mathbf{x}_t^i)).\left(p_t(\mathbf{z}_t|\mathbf{x}_t^i) + p_b(\mathbf{z}_t|\mathbf{x}_t^i)\right) \tag{6}$$

where $p_t(.)$ and $p_b(.)$ are the contributions for the top and bottom hands according to the state hand order, and $p_o(.)$ comes from the hand orientation likelihood. The exclusion principle is enforced through the hand likelihood contributions, which take the form of sums over pixels v in the respective hand boxes r_t and r_b:

$$p_t(\mathbf{z}_t|\mathbf{x}_t^i) = \sum_{v \in r_t} \begin{cases} 1 & \text{if } v \notin r_h \text{ is skin but not moving,} \\ 2 & \text{if } v \notin r_h \text{ is skin and moving,} \\ 0.001 & \text{if } v \in r_h \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where r_h is the head region box, and

$$p_b(\mathbf{z}_t|\mathbf{x}_t^i) = \sum_{v \in r_b} \begin{cases} 1 & \text{if } v \notin r_h \cup r_t \text{ is skin but not moving,} \\ 2 & \text{if } v \notin r_h \cup r_t \text{ is skin and moving,} \\ 0.001 & \text{if } v \in r_h \cup r_t \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Thus each skin or skin-and-motion pixel observation can contribute to one body part only, with the top hand having precedence over the bottom hand. The small contributory factor for pixels falling in the head box ensures that a hand occluding the head is considered as a possibility. Note that under this implementation, the semantics of occlusion are not modelled explicitly but rather in an *ad hoc* manner. The hand orientation likelihood for the top hand is taken from the hand orientation histogram at the top hand position. The bottom hand orientation is not used if it is occluded, since it would probably not be visible in the image. If it is used, then $p_o(.)$ becomes the average of the two likelihoods.

The state propagation density $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$ is a mixture of three functions. The first 10% of state samples are randomly initialised, the next 40% are propagated by random perturbation, and the remainder are propagated according to a simplistic constant-velocity dynamical model by adding the previous spatial hand displacement to $\mathbf{x}_{t-1}^i$ and then adding a random perturbation. The hand precedence state is modified as in [5].

The TE tracker makes an interesting comparison with our BN method because there are several differences in the fundamental approach:

Local vs. global tracking: the CONDENSATION algorithm is generally a localised method in that it is based on state change and dynamical models. Therefore the method is sensitive to initialisation and can permanently lose track of an object if it does not obey the model dynamics. The BN method is global in that it uses skin clusters from the whole image. *Temporal continuity information is used but not relied upon.* Recovery from loss of track is possible at each time frame, therefore the initialisation procedure is not important.

Sparse vs. full density: the CONDENSATION algorithm represents the state density with a sparse sample of particles, while the BN method fully models the state space with an economical number of parameters. Therefore CONDENSATION may miss certain hypotheses if the state space is not adequately sampled.

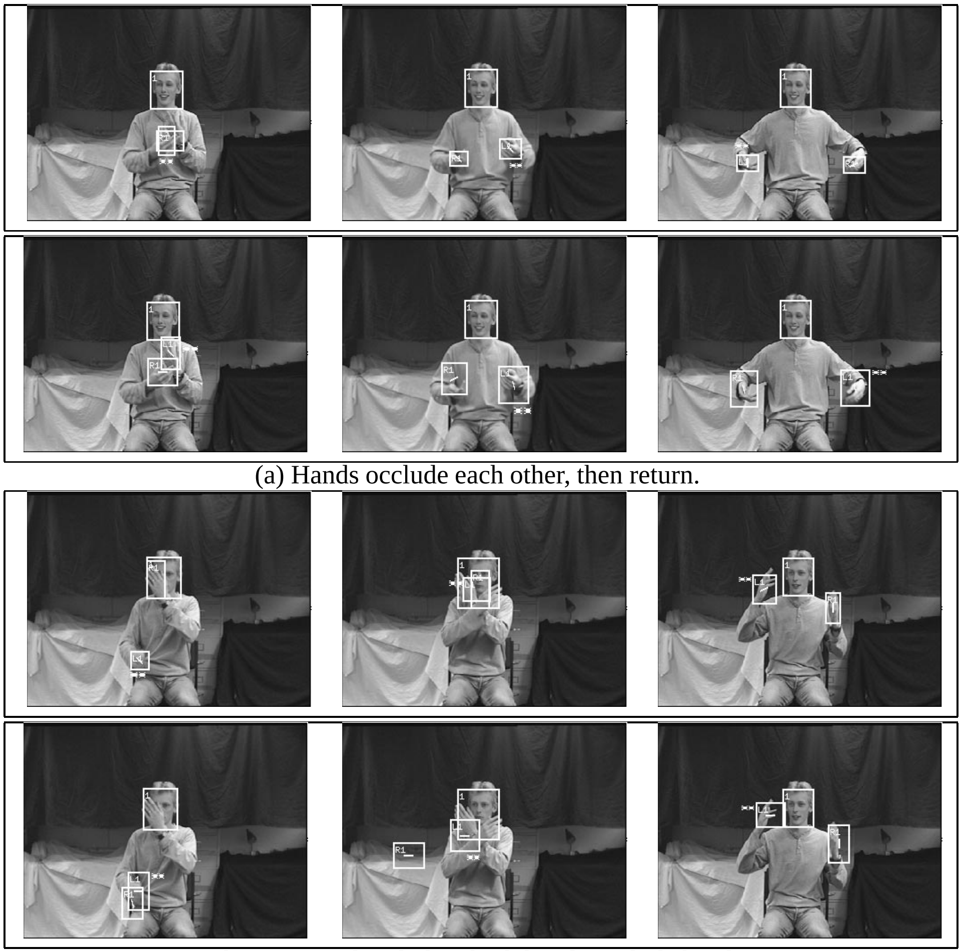
Repeated vs. unique observation: the Bayesian network requires the observations to be used once only to simultaneously consider all possible hypotheses. In comparison, the CONDENSATION algorithm must re-use the observations for each state sample in a hypothesise-and-test fashion which is computationally expensive, and can be wasteful since multiple state samples may be very similar.

Both tracking methods were applied to five sequences consisting of approximately 200 frames each, and involving two different people. The CONDENSATION-based tracker used 2000 state samples, and was initialised by heuristically assuming the hands to be initially found separately in the subject's lap. To illustrate the properties of the two trackers, sample frames from five different occlusion scenarios are shown in Figures 2 and 3. The top row of images in each figure comes from the Bayesian Network tracker, and the bottom row from the TE tracker. In all figures there are three boxes: one for the head, one for the left hand (marked with "**"), and one for the right hand.

In Figure 2(a), the hands occlude each other and then separate. This is to test whether the tracker can obtain the correct hand assignment after occlusion. The BN is able to track both hands during occlusion. Afterwards the tracker assigns the hand positions incorrectly, but after several frames correct assignment is recovered due to the hand orientations. The TE tracker, however, begins and ends with incorrect hand assignment. In Figure 2(b), the right hand first occludes the face, then both hands occlude the face simultaneously. The BN method tracks the hands correctly both during and after occlusion. The TE tracker is unable to detect the head occlusion, and eventually the right hand box locks onto background noise. After occlusion, the TE tracker regains lock.

Figure 3 contains some more challenging examples. In Figure 3(a), the right hand disappears from view entirely and then reappears. Although disappearance of body parts is not modelled explicitly, both trackers are able to recover. During occlusion, the BN tracker assumes that head occlusion has occurred, while the TE tracker catches uncovered pixels at the edge of the visible hand box. Again, the TE tracker maintains *incorrect* hand assignments from the start. In Figure 3(b), the hands occlude and then cross over, with the right hand partially occluding the head afterwards. The BN tracker not only tracks both hands but assigns the hands correctly throughout. The TE tracker copes during occlusion but when the hands cross over, both boxes are assigned incorrectly to the left hand blob. In Figure 3(c), the first two displayed images are consecutive frames labelled t and $t+1$. There is considerable discontinuity in motion between the two frames due to disk swapping during video capture. At time t, the BN tracker is tracking correctly during head occlusion by the right hand. At $t+1$, the hands are found but assigned incorrectly. However, within three frames the tracker has recovered to correctly assign the hands. In contrast, the TE tracker has incorrectly assigned the hands and is unable to track during frame t, instead latching onto some noise. At time $t + 1$, the left hand is found correctly but the right hand is distracted by noise, possibly due to inadequate sampling of the joint state space. After three frames the hands are both found, but incorrectly assigned.

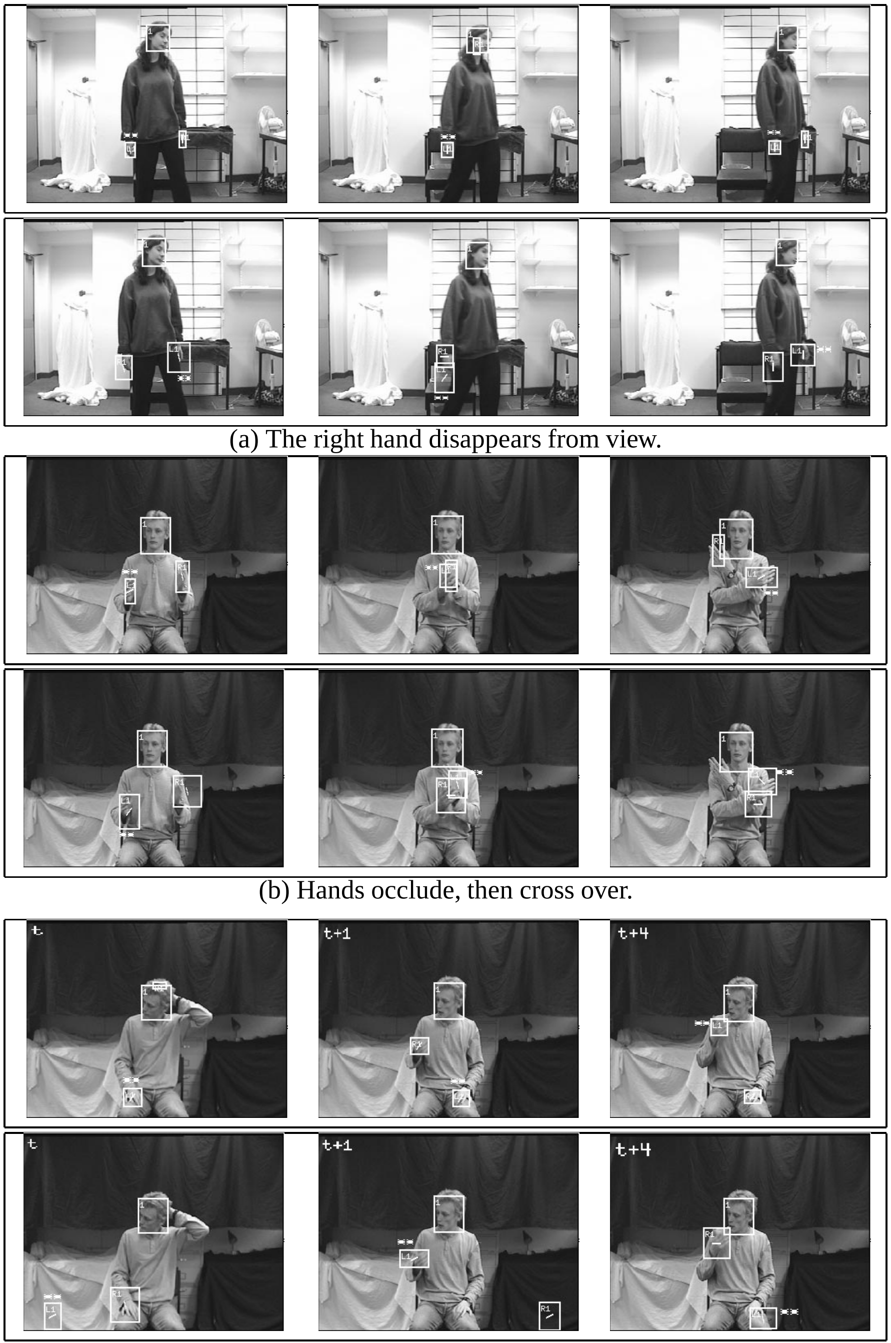
The examples show that the Bayesian network copes well with a variety of complex situations, while the localised sampling method suffers from problems due to inadequate sampling, susceptibility to distractions and lack of high-level constraints. Regarding computational requirements, the BN method required ≈ 1.6 seconds per frame and the TE method required ≈ 24.1 seconds per frame on a PII-330 MHz. The enormous increase in computational expense for the TE method was mainly due to the re-use of observations in statistical sampling, in particular the local hand orientations which require an expensive



(a) Hands occlude each other, then return.

(b) First one hand, then both hands occlude head.

Figure 2: Examples of head and hand tracking results. In each example, the top row is the Bayesian network tracker output, and the bottom row is the CONDENSATION-based tracker output. The left hand box is marked with “**”.



(a) The right hand disappears from view.

(b) Hands occlude, then cross over.

(c) Strongly discontinuous motion between two consecutive frames, t and $t + 1$.

Figure 3: More examples of head and hand tracking results. In each example, the top row is the Bayesian network tracker output, and the bottom row is the CONDENSATION-based tracker output. The left hand box is marked with “**”.

filtering operation. This highlights the important computational advantage of the BN approach: an enormous state space can be fully modelled using efficient computation, while the resources required for particle filtering methods such as CONDENSATION grow exponentially with the state space size and are largely out of the designer's control.

6 Conclusion

A probabilistic reasoning approach to tracking multiple interacting body parts under occlusion and ambiguity using a Bayesian Network has been presented. Bayesian Networks provide a flexible, rigorous and comprehensive framework for incorporating domain knowledge and representing high-level semantics such as occlusion. The tracker considers the whole high-dimensional state space to infer object positions but remains computationally inexpensive, while contemporary methods such as CONDENSATION maintain only a sparse sampling and are commonly expensive to compute. There are several opportunities for extending this work. Firstly, the network probabilities were created by hand. The current network could be used to obtain a training sample of variable values, which could then be employed to re-estimate the network parameters. Secondly, the current network is primarily static in that temporal correlations between variables are largely ignored. The network could be made truly dynamic by adding extra time slices. However, this requires a non-trivial re-structuring of the network at run-time [4]. Investigation is required as to whether the benefits outweigh the extra computational expense.

References

[1] G. Bradski. Computer vision face tracking for use in a perceptual user interface. *Intel Technology Journal*, 2nd Quarter, 1998.

[2] R. Cowell, A. Dawid, S. Lauritzen, and D. Spiegelhalter. *Probabilistic Networks and Expert Systems*. Springer-Verlag, NY, 1999.

[3] D. Heckerman. A tutorial on leraning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Nov 1996.

[4] U. Kjærulff. A computational scheme for reasoning in dynamic probabilistic networks. In *Proc. of Conf. on Uncertainty in AI*, pages 121–129. Morgan Kaufmann, July 1992.

[5] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In *Int. Conf. on Computer Vision*, volume 1, pages 572–578, Corfu, Greece, Sept 1999.

[6] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.

[7] C. Rasmussen and G. Hager. Joint probabilistic techniques for tracking multi-part objects. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 16–21, Santa Barbara, 1998.

[8] J. Sherrah and S. Gong. Tracking discontinuous motion using Bayesian inference. In *Proc. of European Conf. on Computer Vision*, volume 2 of *Springer-Verlag*, pages 150–166, Dublin, Ireland, June 2000.

[9] K. Toyama and E. Horvitz. Bayesian modality fusion: Probabilistic integration of multiple vision algorithms for head tracking. In *Proc. Asian Conf. on Computer Vision*, Jan 2000.

[10] J. Triesch and C. von der Malsburg. Self-organized integration of adaptive visual cues for face tracking. In *Proc. of Int. Conf. on Auto. Face and Gesture Recognition*, pages 102–107, Grenoble, France, Mar 2000. IEEE Press.