Robust Visual Tracking by Integrating Multiple Cues based on Co-inference Learning

YING WU
Department of Electrical & Computer Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208
yingwu@ece.northwestern.edu

THOMAS S. HUANG
Beckman Institute, University of Illinois at Urbana-Champaign, 405 N. Mathews, Urbana, IL 61801
huang@ifp.uiuc.edu

Abstract. Visual tracking can be treated as a parameter estimation problem that infers target states based on image observations from video sequences. A richer target representation would incur better chances of successful tracking in cluttered and dynamic environments, and thus enhance the robustness. Richer representations can be constructed by either specifying a detailed model of a single cue or combining a set of rough models of multiple cues. Both approaches increase the dimensionality of the state space, which results in a dramatic increase of computation. To investigate the integration of rough models from multiple cues and to explore computationally efficient algorithms, this paper formulates the problem of multiple cue integration and tracking in a probabilistic framework based on a factorized graphical model. Structured variational analysis of such a graphical model factorizes different modalities and suggests a *co-inference* process among these modalities. Based on the importance sampling technique, a sequential Monte Carlo algorithm is proposed to provide an efficient simulation and approximation of the *co-inferencing* of multiple cues. This algorithm runs in real-time at around 30Hz. Our extensive experiments show that the proposed algorithm performs robustly in a large variety of tracking scenarios. The approach presented in this paper has the potential to solve other problems including sensor fusion problems.

Keywords: Visual tracking, sequential Monte Carlo, importance sampling, co-inference, factorized graphical model, variational analysis

1. Introduction

With the rapid enhancement of computational power provided by computer hardware, computers are more likely to afford some visual capacities. Recent years have witnessed an expeditious development of the research and applications of visual surveillance and vision-based interfaces, in which visual tracking plays an important role. Aiming at developing more natural and non-invasive human computer interfaces and recognizing human actions visually, tremendous research efforts have been devoted to visual tracking and analysis of human movements (Gavrila, 1999; Pavlović et al., 1997; Wu and Huang, 2001a).

One of the purposes of visual tracking is to infer the *states* of the targets from image sequences. Besides 2D positions, visual tracking is also expected to recover other states, such as poses, articulations or deformations, depending on different applications. Although the tracking problem is well formulated in the research of control theory and signal processing, visual tracking involves many fundamental research problems in object representations, image analysis and matching. Since the target states are hidden and can only be inferred from observable visual features, two difficulties confronts visual tracking: evaluating state hypotheses on the observed image evidence and searching the state space.

Bottom-up and *top-down* approaches are two kinds of methodologies for the visual tracking problem. *Bottom-up* approaches generally tend to reconstruct the target states by analyzing the image contents. For example, reconstructing a parametric shape by curve fitting. In contrast, top-down approaches generate and evaluate a set of state hypotheses based on target models. Tracking

is achieved by evaluating and verifying these hypotheses on image observations. Certainly, these two approaches could be combined.

Bottom-up methods might be computationally efficient, yet the robustness largely depends on the ability of image analysis, because grouping, tracing and fitting image pixels could be overwhelmed by image clutters and noise. On the other hand, top-down approaches depend less on image analysis, because the target hypotheses serve as strong constraints for analyzing images. But the performance of the top-down approaches are largely determined by the methods of generating and verifying hypotheses. To achieve robust tracking, a large number of hypotheses may be maintained so that more computation would be involved for evaluating them. The combination of these two methodologies could keep the robustness but reduce the computation.

Visual tracking techniques generally have four elements, the *target representation*, the *observation representation*, the *hypotheses generating*, and the *hypotheses evaluation*, which roughly characterize tracking performance and limitations.

To discriminate the target from other objects, the target representation, denoted by $\mathbf{X}$, which could include different modalities such as shape, color, appearance, and motion, characterizes the target in a *state space* either explicitly or implicitly. Although finding the representations for targets is a fundamental problem in computer vision, visual tracking research generally employs concise representations to facilitate computational efficiency. For example, parameterized shapes (Isard and Blake, 1996; Isard and Blake, 1998b), and color distributions (Comaniciu et al., 2000; Raja et al., 1998; Toyama et al., 1999; Wu and Huang, 2000) are often used as target representations. To provide a more constrained description of the target, some methods employ both shape and color (Azoz et al., 1998; Birchfield, 1998; Isard and Blake, 1998b; Rasmussen and Hager, 1998; Wren et al., 1997; Toyama and Wu, 2000). Obviously, a unique characterization of the target would be quite helpful to visual tracking, but it will involve high dimensionality. To add uniqueness to the target representation, many methods even employ image appearances, such as image templates (Hager and Belhumeur, 1996; Li and Chellapa, 2000; Tao et al., 2000) or eigenspace representation (Black and Jepson, 1996), as target representations. For example, if you know a person, it would be a bit easier to track this person in a crowd. In addition, motion can also be taken into account in target representations, since different objects can be discriminated by the differences of their motions. On the other hand, if two objects share the same representation, it would be difficult to correctly track either of them when they are close in the *state space*, if there is no prior from the dynamics of the targets' movements.

Closely related to the target representation is the observation representation, denoted by $\mathbf{Z}$, which defines the image evidence, i.e., the image features observed. For example, if the target is represented by its contour shape, the corresponding image edges are expected to be observed in images. If the target is characterized by its color appearances, certain color distribution patterns in images can be used as the observations of the target.

The hypotheses evaluation calculates the matching between state hypotheses and image observations. We need to measure the likelihood of the image observations given state hypotheses, i.e., $p(\mathbf{Z}|\mathbf{X})$, so as to infer the posterior $p(\mathbf{X}|\mathbf{Z})$ of a hypothesis given a certain image observation in the MAP estimation framework. However, the evaluation would be quite challenging when evaluating a shape hypothesis on an image with clutters. Although some analytical results were reported in (Blake and Isard, 1998), many current tracking methods take *ad hoc* measurements.

The hypothesis generation, denoted by $p(\mathbf{X}_t|\mathbf{X}_{t-1})$, produces new state hypotheses based on old estimates of the target state, implying the evolution of the target's dynamic processes. Target's dynamics can be embedded in such a predicting process. At a certain time instant, the target state is a random vector. The *a posteriori* probability distribution of the target state given the observation history changes with time. Therefore, the tracking problem can be viewed as a problem of propagating conditional probability densities. The target state at a certain time instant can

be calculated through estimating the conditional probability density of it. The Kalman filtering technique gives a classic example of hypotheses generation under Gaussian assumptions, due to which the densities are characterized and parameterized by their means and covariances. Thus, the hypothesis generation characterizes the search range and confidence level of the tracking. On the other hand, if the Gaussian assumption does not hold, which is very likely in image clutters, we can represent the posterior densities in non-parametric forms. In this circumstance, the hypotheses generation can be viewed as a evolution process of a set of hypotheses or state samples or particles, which facilitates a Monte Carlo approach for tracking. The CONDENSATION (Blake and Isard, 1998) algorithm is one such example.

Using a rough target model would not be robust. For example, if we use an ellipse to model the head, the visual tracking might be unstable in a cluttered environment, e.g., when the head moves in front of a book shelf, since the false image edges incurred by the clutter would likely distract the tracker. Thus, to enhance the robustness, a richer target representation should be employed, since it brings more uniqueness. There are two kinds of ideas. One approach is to construct and use a detailed target model. For example, a B-spline shape model can be used to accurately describe a contour, based on which excellent contour tracking results have been reported (Blake and Isard, 1998). The other approach is to combine several rough representations or models of different cues. For example, we can simultaneously use a rough shape model and a rough color model to represent the head. Would the combination of multiple rough models result in robust tracking results? If so, how to integrate multiple cues and rough models? How do these different cue models interact with each other? This paper will try to answer these questions.

This paper formulates the problem of integrating multiple cues for robust tracking as the probabilistic inference problem of a factorized graphical model. To analyze this complex graphical model, a *variational method* is taken to approximate the Bayesian inference. Interestingly, the analysis reveals a *co-inference* phenomenon of multiple modalities, which illustrates the interactions among different cues, i.e., one cue can be inferred iteratively by the other cues. Based on this, this paper presents an efficient sequential Monte Carlo tracking algorithm to integrate multiple visual cues, in which the *co-inference* of different modalities is approximated.

In Section 2, we will give a brief overview of the research of multiple cue integration in the context of robust tracking. Then the factorized graphical model used in our tracking formulation will be presented in Section 3; the *co-inference* phenomenon will be analyzed and explained in this section as well. Section 4 will describe different techniques in sequential Monte Carlo approaches for tracking problems. Based on importance sampling techniques, our proposed approach of the *co-inference* will be presented in Section 5, and the details of our tracking implementation and experiments will be described in Section 7. Section 8 will conclude the paper by discussing the proposed approach and pointing out some possible directions for future investigations.

2. Multiple Cue Integration

We often notice that a state hypothesis of a richer target representation, either a detailed model or a combination of multiple rough models, would have better opportunities to be verified on various image observations. For example, combining the color appearance of the target can largely enhance the robustness of contour tracking against heavily cluttered backgrounds, and integrating shape and color representations could result in better tracking performance against color distracters.

In addition to effective hypothesis evaluation, integrating multiple cues would reduce tracker's dependency on accurate dynamic models of the targets. Dynamics models play an important role in tracking since they provide predictions to reduce search and computations. In many cases, the parameters of dynamic models are specified in advance, or learned by training sequences.

However, if the parameters are not properly set, the tracker would be under large risks of failure. It is desirable to develop robust trackers that work with weak dynamic models.

Interestingly, the integration and interaction of multiple cues for tracking would not require accurate dynamic models. Here is the intuition. Suppose we represent the target by two modalities: shape and color appearance. The two modalities have their own dynamic models, which means that the target is deformable, and the lighting could change, but it is difficult to know in advance about how the shape will deform and how the lighting will change. Therefore, we can only assume very rough dynamic models as approximations. However, such rough dynamic models will be sometimes violated such that the predictions based on the dynamic models could largely deviate and fail the tracker, if the two modalities are treated separately. Our main idea is to let the two modalities interact with each other. For example, if the shape changes very little but the lighting changes a lot, the estimation of the color appearance can be fulfilled by taking advantage of shape estimates, instead of relying on the predictions from the dynamics of color appearances. Symmetrically, if the lighting changes very slowly but the target deforms a lot, the deformation can be more robustly localized with the help of the color appearance estimates, instead of taking a strong prediction prior from the deformation dynamics. Naturally, we shall ask what if the changes of both deformation and lighting are quite large? The problem becomes a sort of *untrackable* problem if no accurate dynamic model is available. Fortunately, we can still approach it by taking the estimation that maximizes the joint probability of both modalities, and find the most likely state estimates. Detailed formulation and analysis will be given later in this paper.

Multiple cue integration can be done at the level of both observation representation and object model. At the observation level, some approaches combine the measurements of the multiple modalities for each hypothesis (Azoz et al., 1998; Birchfield, 1998). Although robust to some extent, the methods of combining the likelihood measurements of different sources are often *ad hoc*. In addition, to integrate shape and color, many tracking algorithms assume fixed target color appearances (Azoz et al., 1998; Darrell et al., 1998; Isard and Blake, 1998b; Toyama and Wu, 2000) to enable efficient color segmentation. However, such an assumption is often invalid in practice. Instead of assuming a fixed color representation, non-stationary color tracking methods (Raja et al., 1998; Wu and Huang, 2000) adapt the color changes and update the color models. At the object representation level, some methods also includ the color modality in the target representation (Bregler, 1997; Rasmussen and Hager, 1998; Wren et al., 1997), in which a multivariate Gaussian can be used to represent both color and motion parameters. Obviously, tracking both shape and color simultaneously would be a formidable problem, since it increases the dimensionality of the state space. In addition, the interaction of multiple modalities is interesting and important for the robustness. A switching scheme can be use to coordinate different trackers that track different modalities (Toyama and Hager, 1996; Toyama and Wu, 2000). Generally, different modalities are updated sequentially in these methods. However, a more profound and systematic investigation of the interaction of multiple modalities is desirable. This paper tries to investigate the relationship among different modalities for robust visual tracking, and to identify an efficient way to facilitate simultaneously tracking of these modalities.

3. Graphical Models for Tracking

In this section, we formulate the visual tracking problem in a probabilistic framework. The integration of multiple cues is characterized by a factorized graphical model, and we use the variational analysis approach to approximate the probabilistic inference.

Following the notations of Isard and Blake (Isard and Blake, 1996; Blake and Isard, 1998), we denote the target states and image observations by $\mathbf{X}_t$ and $\mathbf{Z}_t$, respectively. The history of the

states and measurements are denoted by $\underline{X}_t = (\mathbf{X}_1, \ldots, \mathbf{X}_t)$ and $\underline{Z}_t = (\mathbf{Z}_1, \ldots, \mathbf{Z}_t)$. The tracking problem can be formulated as an inference problem with the prediction prior $p(\mathbf{X}_{t+1}|\underline{Z}_t)$. We have

$$p(\mathbf{X}_{t+1}|\underline{Z}_{t+1}) \propto p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1})p(\mathbf{X}_{t+1}|\underline{Z}_t)$$
$$p(\mathbf{X}_{t+1}|\underline{Z}_t) = \int p(\mathbf{X}_{t+1}|\mathbf{X}_t)p(\mathbf{X}_t|\underline{Z}_t)d\mathbf{X}_t$$

where $p(\mathbf{Z}_t|\mathbf{X}_t)$ represents the *measurement* or *observation* likelihood, and $p(\mathbf{X}_{t+1}|\mathbf{X}_t)$ is the dynamic model.

The probabilistic formulation of the tracking problem can be represented by the graphical model shown in Figure 1, where the $\mathbf{X}$ nodes are hidden states and $\mathbf{Z}$ nodes are observations. This is similar to the Hidden Markov Model (Rabiner, 1989). At time t, the observation $\mathbf{Z}_t$ is independent of previous states $\underline{X}_{t-1}$ and previous observations $\underline{Z}_{t-1}$, given current state $\mathbf{X}_t$, i.e., $p(\mathbf{Z}_t|\underline{X}_t, \underline{Z}_{t-1}) = p(\mathbf{Z}_t|\mathbf{X}_t)$, and the states have a first order Markov property, i.e., $p(\mathbf{X}_t|\underline{X}_{t-1}) = p(\mathbf{X}_t|\mathbf{X}_{t-1})$.

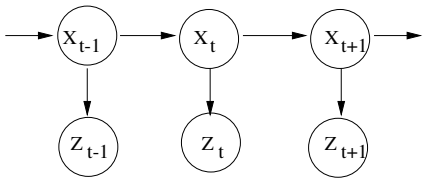

Figure 1. The tracking problem can be represented by a graphical model, similar to the Hidden Markov Model.

Based on this graphical model, the tracking problem can be approached by probabilistic inference techniques. However, when the dimensionality of the hidden states increases, the inference and learning would become difficult due to the dramatic increase of computation. Fortunately, a distributed state representation based on factorized graphic models can largely ease this difficulty by decoupling the dynamics of different subsets of hidden states. Combining a set of rough models for different cues can be formulated by such factorized graphical models. For example, target states can be decomposed into shape states and color states as shown in Figure 2(a). In addition, the observation could also be separated into $\mathbf{Z}_t^s$ and $\mathbf{Z}_t^c$ for shape and color respectively in Figure 2(b). Each observation depends on both color and shape states.

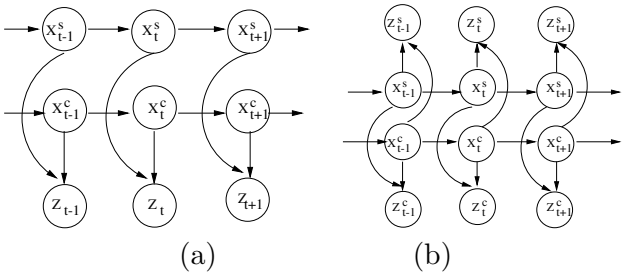

Figure 2. Factorized Graphical Models: (a) The states of the target can be decomposed into shape states $\mathbf{X}_t^s$ and color states $\mathbf{X}_t^c$ in a factorized graphical model. (b) The observation could also be separated into $\mathbf{Z}_t^s$ and $\mathbf{Z}_t^c$.

Due to the complex structure of the densely connected factorized network in Figure 2, the exact inference would be formidable. One approach to this problem is based on statistical sampling methods, such as Gibbs sampling. Another approach is an analytical way through probabilistic variational analysis. The basic idea is to approximate the posterior probability $p(\underline{X}_t|\underline{Z}_t)$ of the hidden states by a tractable distribution $Q(\underline{X}_t)$ which has good analytical properties. The optimal model parameters as well as the variational parameters would be found by minimizing the discrepancy between these two distributions. A lower bound on the log likelihood $\log P(\underline{Z}_t)$ can

be achieved by such an approximation (Saul and Jordan, 1996; Ghahramani, 1995; Ghahramani and Jordan, 1997; Jordan et al., 2000):

$$\log P(\underline{Z}_t) \geq \sum_{X_t} Q(\underline{X}_t) \log \frac{P(\underline{X}_t, \underline{Z}_t)}{Q(\underline{X}_t)} \tag{1}$$

$$KL(Q||P) = \sum_{X_t} Q(\underline{X}_t) \log \frac{Q(\underline{X}_t)}{P(\underline{X}_t|\underline{Z}_t)} \tag{2}$$

Generally, we can choose $Q(\cdot)$ to have a simpler structure by eliminating some of the dependencies in the original factorized model, while minimizing the Kullback-Leibler divergence between $P(\cdot)$ and $Q(\cdot)$ in equation 2. The analysis can be achieved by a *structured variational inference* technique. The basic idea is to uncouple the Markov chains and replace the true observation probability $p(\mathbf{Z}_t|\mathbf{X}^m)$ of each hidden state by distinct variational parameters $h^m_{x_t}$, as shown in Figure 3. The structured variational model subjected to a set of variational parameters has much simpler structure than the original factorized graphical model. Therefore, the inference problem is easier for the structured variational model. The inference of the original factorized model $p(\underline{X}_t|\underline{Z}_t)$ is expected to be approximated by the inference of variational inference $Q(\underline{X}_t|\theta)$ when optimal variational parameters θ are identified. Here, $\theta = \{h^m_{x_t}\}$.

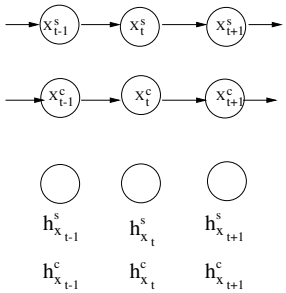

Figure 3. Structured Variational Model: uncoupling the Markov chains and replacing the true observation probability of each hidden state by distinct variational parameters $h^m_{x_t}$.

Suppose the target state includes M modalities, for the above structured variational model in Figure 3, we can write the distribution of the hidden states given a set of variational parameters $Q(\underline{X}_t|\theta)$. Here, $h^m_{x_t}$ are the variational parameters.

$$\begin{aligned} Q(\underline{X}_t|\theta) &= \frac{1}{Z_Q} \prod_{m=1}^{M} Q(\mathbf{X}^m_1|\theta) \prod_{t=2}^{T} Q(\mathbf{X}^m_t|\mathbf{X}^m_{t-1}, \theta) \\ &= \frac{1}{Z_Q} \prod_{m=1}^{M} h^m_{x_1} \pi^m_0 \prod_{t=2}^{T} h^m_{x_t} p(\mathbf{X}^m_t|\mathbf{X}^m_{t-1}) \end{aligned}$$

where M is the number of modalities or factorized Markov chains, $\mathbf{X}^m_t$ is the state of the m-th modality at time frame t, $h^m_{x_t}$ is the variational parameter of the hidden state of the m-th chain at time t, π^m_0 is the initial probability of the hidden states of the m-th chain, Z_Q is a normalization constant. Although general continuous analysis of this approach is unavailable, a simplified case of discrete hidden states and linear discrete observations can be analyzed (Ghahramani and Jordan, 1997). Under the assumption of linear observations, a set of fixed point equations for $h^m_{x_t}$ to minimize $KL(Q||P)$ can be obtained (Ghahramani and Jordan, 1997):

$$\widetilde{h^m_{x_t}} = exp\left\{ W^{m\prime} C^{-1} \left[\mathbf{Z}_t - \sum_{k \neq m}^{M} W^k E\left[\mathbf{X}^m_t|\theta, \underline{Z}_t\right] \right] - \frac{1}{2}\Delta^m \right\} \tag{3}$$

where W^m is an observation matrix and C is the covariance matrix, since the above analytical derivation assumes a linear observation model, and Δ^m is the vector of the diagonal elements of $W^{m\prime}C^{-1}W^m$. The details of the derivation can be found in the appendix of this paper. To make Equation 3 clearer, we can write this set of fixed point equations by

$$\widetilde{h^m_{x_t}} = \mathcal{G}(\mathbf{Z}_t, \{E[\mathbf{X}^n_t|\underline{Z}_t, \theta] : \forall n \neq m\}) \tag{4}$$

where $\mathcal{G}(\cdot,\cdot)$ is a function, and $E[\mathbf{X}^n_t|\underline{Z}_t, \theta] \equiv \langle \mathbf{X}^n_t \rangle$ is the expectation of the hidden state $\mathbf{X}^n_t$ at the n-th uncoupled Markov chain, based on the variational parameters h^n. Using these variational parameters, a new set of expectations for the hidden states $\langle \mathbf{X}^m_t \rangle$ will be fed back into Equation 4, which can be solved iteratively. It is similar to the EM algorithm (Dempster et al., 1977). To make it clear, we can explicitly write up in Equation 5 the fixed point equations of Equation 4 for the case of two modalities, for example, shape and color:

$$\begin{cases} \widetilde{h^s_{x_t}} &= \mathcal{G}(\mathbf{Z}_t, E[\mathbf{X}^c_t|\underline{Z}_t, \theta]) \\ \widetilde{h^c_{x_t}} &= \mathcal{G}(\mathbf{Z}_t, E[\mathbf{X}^s_t|\underline{Z}_t, \theta]) \end{cases} \tag{5}$$

where $\mathbf{X}^s_t$ is the shape state, $\mathbf{X}^c_t$ is the color state, and $h^s_{x_t}$ and $h^c_{x_t}$ represent the shape and color variational parameters, respectively. Interestingly, the variational parameters $h^s_{x_t}$ and $h^c_{x_t}$ act as the roles of the observation probabilities $p(\mathbf{Z}_t|\mathbf{X}^s_t)$ and $p(\mathbf{Z}_t|\mathbf{X}^c_t)$ in the original factorized model.

It should be noticed that the original densely connected graphical model is uncoupled in the structured variational model. The hidden states of each uncoupled Markov chain could be estimated separately, given the set of variational parameters. The estimation of the variational parameters for one chain depends on the hidden states of the other chains. Such phenomenon becomes quite clear in Equation 5, where in the iteration of the fixed point equations, the shape state estimation $E[\mathbf{X}^s_t|\underline{Z}_t, \theta]$ is used to calculate the variational parameters of color modality $\widetilde{h^c_{x_t}}$, and the color state estimation $E[\mathbf{X}^c_t|\underline{Z}_t, \theta]$ is used to calculate the variational parameters of shape modality $\widetilde{h^s_{x_t}}$. We call such an interesting phenomenon *co-inference* (Wu and Huang, 2001b), since the parameters of one modality could be inferred iteratively by other modalities.

The structured variational analysis of the factorized model in Figure 2 is meaningful for the problem of multiple cue integration, since it reveals the interactions among different modalities. It thus suggests an efficient approach to track multiple cues, which will be presented in section 5.

4. Monte Carlo Tracking

As described in previous sections, the visual tracking problem can be formulated in a probabilistic framework by representing tracking as a process of conditional probability density propagation. Although analytical solutions of this inference problem are generally intractable for non-Gaussian cases, Monte Carlo methods offer a powerful means to approximate the probabilistic inference and to characterize the evolution processes of the dynamic systems.

In statistics, sampling techniques are widely used to approximate a complex probability density. Sequential Monte Carlo methods for dynamic systems are also studied in the area of statistics (Liu and Chen, 1998; Liu et al., 2000; Doucet et al., 2000). A set of weighted random samples $\{(s^{(n)}, \pi^{(n)})\}, n = 1, \ldots, N$ is *properly weighted* with respect to the distribution $f(\mathbf{X})$ if for any integrable function $h(\cdot)$,

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N} h(s^{(n)})\pi^{(n)}}{\sum_{n=1}^{N} \pi^{(n)}} = E_f[h(\mathbf{X})] \tag{6}$$

where $E_f[h(\mathbf{X})]$ is the expectation of $h(\mathbf{X})$. In this sense, the distribution $f(\mathbf{X})$ is approximated by a set of discrete random samples $s^{(n)}$, each having a probability proportional to its weight

$\pi^{(n)}$. We can use the sampling methods for the tracking problem: since the posterior $p(\mathbf{X}_t|\underline{Z}_t)$ is represented by a set of weighted random samples $\{(s_t^{(n)}, \pi_t^{(n)})\}$, this sample set will evolve into a new sample set $\{(s_{t+1}^{(n)}, \pi_{t+1}^{(n)})\}$ representing the posterior $p(\mathbf{X}_{t+1}|\underline{Z}_{t+1})$ at time $t+1$. Thus, tracking can be characterized by the evolution of such a set of weighted samples in the state space.

4.1. Factored Sampling

To represent the posterior $p(\mathbf{X}_t|\underline{Z}_t)$, a set of random samples $\{\mathbf{X}_t^{(n)}, n = 1, \ldots, N\}$ can be drawn from a prediction prior $p(\mathbf{X}_t|\underline{Z}_{t-1})$, and weighted by their image measurements, i.e., $\pi_t^{(n)} = p(\mathbf{Z}_t|\mathbf{X}_t = \mathbf{X}_t^{(n)})$, such that the posterior $p(\mathbf{X}_t|\underline{Z}_t)$ is represented by a set of weighted random samples $\{s_t^{(n)}, \pi_t^{(n)}\}$. This sampling scheme is called *factored sampling* (Isard and Blake, 1996; Blake and Isard, 1998; Liu and Chen, 1998; Liu et al., 2000). It can be shown that such a sample set is properly weighted. This sample set will evolve to a new sample set at time $t+1$ and the new sample set $\{s_{t+1}^{(n)}, \pi_{t+1}^{(n)}\}$ represents the posterior $p(\mathbf{X}_{t+1}|\underline{Z}_{t+1})$ at time $t+1$. This is the sequential Monte Carlo method employed in Condensation algorithm (Isard and Blake, 1996; Blake and Isard, 1998; Isard and Blake, 1998a).

Condensation achieved quite robust tracking results. The robustness of the sequential Monte Carlo tracking is due to the maintenance of a pool of hypotheses. Since each hypothesis needs to be evaluated and associated with a likelihood value, the computational cost mainly comes from the image measurement processes. More samples results in more accurate tracking results but slower tracking speed. Consequently, the number of samples becomes an important factor in Monte Carlo based tracking. Unfortunately, when the dimensionality of the state space increases, the number of samples increases exponentially.

This phenomenon has been observed and different methods have been taken to reduce the number of samples. A semi-parametric approach was taken in (Cham and Rehg, 1999), which retained only the modes (or peaks) of the probability densities, and represented the local neighborhood surrounding each mode as a Gaussian distribution. This approach eliminated the need for a large number of samples to represent the distribution around each mode. In addition, different sampling techniques were also investigated to reduce the number of samples. In (MacCormick and Isard, 2000), a partitioned sampling scheme was proposed to track articulated objects. It was basically a hierarchical method to generate the hypotheses. Similar approach was taken in (Tao et al., 1999) to track multiple objects. In (Deutscher et al., 2000), an annealed particle filtering scheme was taken to search the global maximum of the posterior probability density. In (MacCormick and Blake, 1999), an exclusion scheme was proposed to approach the occlusion problem in multiple targets tracking.

4.2. Importance Sampling

In practice, it might be difficult to draw random samples directly from a distribution $f(\mathbf{X})$. Instead, samples can be drawn from another distribution $g(\mathbf{X})$, called the importance function, but sample weights should be properly adjusted. This is the basic idea of the *importance sampling* technique, which is a powerful tool in statistics (Tanner, 1993; Liu et al., 2000). Isard and Blake introduced this technique to visual tracking in the ICondensation algorithm (Isard and Blake, 1998b). Instead of sampling the prediction prior $f_t(\mathbf{X}_t) = p(\mathbf{X}_t|\underline{Z}_{t-1})$ of the dynamics, ICondensation sampled an outside importance prior $g_t(\mathbf{X}_t)$ which was induced by color segmentation.

In important sampling, samples $s^{(n)}$ are drawn from an importance function $g(\mathbf{X})$ while the weights are compensated by

$$\pi^{(n)} = \frac{f(s^{(n)})}{g(s^{(n)})}\tilde{\pi}^{(n)},$$

where $\tilde{\pi}^{(n)}$ are the un-compensated weights associated with the sampling of $g(\mathbf{X})$. It can be proved that the sample set $\{s^{(n)}, \pi^{(n)}\}$ is still *properly weighted* with respect to $f(\mathbf{X})$. This is illustrated in Figure 4.

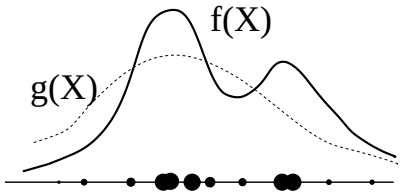

Figure 4. Importance sampling. Samples that are drawn from another distribution $g(\mathbf{X})$ but with adjusted weights can still be used to represent density $f(\mathbf{X})$.

To employ the importance sampling technique in dynamic systems, we let $f_t(\mathbf{X}_t^{(n)}) = p(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\underline{Z}_{t-1})$, where $f_t(\cdot)$ is the tracking prior, i.e., a prediction density. So, when we want to infer the posterior $p(\mathbf{X}_t|\underline{Z}_t)$, we can draw random samples from another distribution $g_t(\mathbf{X}_t)$, instead of the prediction prior density $f_t(\mathbf{X}_t)$. To evaluate $f_t(\mathbf{X}_t)$, we have:

$$\begin{aligned} f_t(\mathbf{X}_t^{(n)}) &= p(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\underline{Z}_{t-1}) \\ &= \sum_{k=1}^{N} \pi_{t-1}^{(k)} p(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\mathbf{X}_{t-1} = \mathbf{X}_{t-1}^{(k)}) \end{aligned}$$

Thus, to approximate a posterior $p(\mathbf{X}_t|\underline{Z}_t)$, samples $s^{(n)}$ can be drawn from another outside importance source $g_t(\mathbf{X}_t)$, and the weight of each sample is:

$$\pi_t^{(n)} = \frac{f_t(s_t^{(n)})}{g_t(s_t^{(n)})} p(\mathbf{Z}_t|\mathbf{X}_t = s_t^{(n)}) \tag{7}$$

where $f_t(s_t^{(n)}) = p(\mathbf{X}_t = s_t^{(n)}|\underline{Z}_{t-1})$. Details can be found in (Isard and Blake, 1998b). We should notice here that in order to sample from $g_t(\mathbf{X}_t)$ instead of $f_t(\mathbf{X}_t)$, both $f_t(s_t^{(n)})$ and $g_t(s_t^{(n)})$ should be evaluatable.

5. Co-inference Tracking

The structured variational analysis of the factorized graphical model in Section 3 suggests a way to uncouple the multiple modalities. However, the analytical analysis does not offer a direct implementation of the *co-inference* for visual tracking. In this section, we present an efficient algorithm to simulate and approximate the *co-inference* of the variational analysis based on statistical sampling and sequential Monte Carlo techniques.

Let $s_t^{(n)} = (s_t^{s,(n)}, s_t^{c,(n)})$ denote the n-th sample of the target state at time t, where $s_t^{s,(n)}$ and $s_t^{c,(n)}$ represent shape state and color state of the sample, respectively. $\pi_t^{s,(n)}$, $\pi_t^{c,(n)}$, and $\pi_t^{(n)}$ denote the sample weights based on shape observation, color observation and a combination of shape and color observation, respectively. At time t, we have a set of samples with

weights $\{(s_t^{s,(n)}, s_t^{c,(n)}, \pi_t^{s,(n)}, \pi_t^{c,(n)}, \pi_t^{(n)}), n = 1, \ldots, N\}$. To generate the samples at time $t+1$, i.e., $\{(s_{t+1}^{s,(n)}, s_{t+1}^{c,(n)}, \pi_{t+1}^{s,(n)}, \pi_{t+1}^{c,(n)}, \pi_{t+1}^{(n)}), n = 1, \ldots, N\}$, an iterative procedure is shown in Figure 5.

Generate $\{(s_{t+1}^{s,(n)}, s_{t+1}^{c,(n)}, \pi_{t+1}^{s,(n)}, \pi_{t+1}^{c,(n)}, \pi_{t+1}^{(n)})\}$ from $\{(s_t^{s,(n)}, s_t^{c,(n)}, \pi_t^{s,(n)}, \pi_t^{c,(n)}, \pi_t^{(n)})\}$,$n = 1, \ldots, N$:

//Step(0): Initialization
$s_{(0)}^{(\cdot)} = s_t^{(\cdot)}$; $\quad \pi_{(0)}^{*,(\cdot)} = \pi_t^{*,(\cdot)}$;

`for` $k = 0 : K-1$
  //Step(1): Shape samples generating
  $s_{(k+1)}^{s,(\cdot)} =$ `I_Samping(`$\{(s_{(k)}^{s,(\cdot)}, \pi_{(k)}^{c,(\cdot)})\}$`);`

  //Step(2): Shape observation
  $\pi_{(k+1)}^{s,(\cdot)} =$`Shape_Obsrv(`$s_{(k+1)}^{s,(\cdot)}$`);`

  //Step(3): Color samples generating
  $s_{(k+1)}^{c,(\cdot)} =$`I_Sampling(`$\{(s_{(k)}^{c,(\cdot)}, \pi_{(k+1)}^{s,(\cdot)})\}$`);`

  //Step(4): Color observation
  $\pi_{(k+1)}^{c,(\cdot)} =$`Color_Obsrv(`$s_{(k+1)}^{c,(\cdot)}$`);`
`end`

$s_{t+1}^{(\cdot)} = s_{(K)}^{(\cdot)}$; $\quad \pi_{t+1}^{*,(\cdot)} = \pi_{(K)}^{*,(\cdot)}$; $\quad \pi_{t+1}^{(\cdot)} = \pi_{t+1}^{s,(\cdot)} \pi_{t+1}^{c,(\cdot)}$;

Figure 5. Co-inference tracking algorithm I: *top-down*

The basic idea behind the above iteration is that one modality receives importance priors from other modalities such that the *co-training* among all the modalities will tend to maximize the likelihood. Specifically, at first shape samples are drawn according to the color importance prior based on importance sampling, i.e., shape samples are drawn from $g_s \sim \{(s_t^{s,(n)}, \pi_t^{c,(n)})\}$ instead of $f_s \sim \{(s_t^{s,(n)}, \pi_t^{s,(n)})\}$. Since the clutter could also incur high shape measurements, sampling only according to the shape prediction prior (as in CONDENSATION) is not appropriate to handle clutter backgrounds, especially when a rough shape representation is taken. On the other hand, sampling according to the color importance prior would largely ease this difficulty, since the samples with higher color measurements would have higher probability to propagate. Weight corrections are:

$$\begin{aligned} \pi_t^{s,(n)} &= \frac{f_s(s_t^{s,(n)})}{g_s(s_t^{s,(n)})} p(\mathbf{Z}_t | \mathbf{X}_t = s_t^{(n)}) \\ f_s(s_t^{s,(n)}) &= \sum_{k=1}^{N} \pi_{t-1}^{s,(k)} p(\mathbf{X}_t^s = s_t^{s,(n)} | \mathbf{X}_{t-1}^s = s_{t-1}^{s,(k)}) \end{aligned}$$

Symmetrically, color samples are then drawn according to the shape importance prior based on importance sampling, i.e., color samples are drawn from $g_c \sim \{(s_t^{c,(n)}, \pi_t^{s,(n)})\}$ instead of $f_c \sim \{(s_t^{c,(n)}, \pi_t^{c,(n)})\}$. This step would let color samples with higher shape measurements to have better

chances to propagate to the next time step.

$$\pi_t^{c,(n)} = \frac{f_c(s_t^{c,(n)})}{g_c(s_t^{c,(n)})} p(\mathbf{Z}_t|\mathbf{X}_t = s_t^{(n)})$$
$$f_c(s_t^{c,(n)}) = \sum_{k=1}^{N} \pi_{t-1}^{c,(k)} p(\mathbf{X}_t^c = s_t^{c,(n)}|\mathbf{X}_{t-1}^c = s_{t-1}^{c,(k)})$$

The above two steps approximate the *co-inference*. The iteration would increase the likelihood of observations. For simplicity, we let $\pi_t^{(n)} = \pi_t^{s,(n)}\pi_t^{c,(n)}$, and the estimates of the shape and color states are given by:

$$\bar{\mathbf{X}}_t^s = \frac{\sum_{n=1}^{N} s_t^{s,(n)}\pi_t^{(n)}}{\sum_{n=1}^{N} \pi_t^{(n)}}; \quad (8)$$

$$\bar{\mathbf{X}}_t^c = \frac{\sum_{n=1}^{N} s_t^{c,(n)}\pi_t^{(n)}}{\sum_{n=1}^{N} \pi_t^{(n)}}. \quad (9)$$

Our approach is different from the ICondensation algorithm in (Isard and Blake, 1998b). Their method assumes a fixed color distribution as the importance prior, while our approach can track both shape and color due to the *co-inference* between them. In the situation of fixed color dynamics, our approach would be similar to their method.

The above algorithm takes the *top-down* approach for both shape and color by generating samples in the joint shape and color state space. However, we notice that it would be more efficient to combine the *top-down* and *bottom-up* approaches, since the color state can be easily estimated by taking a bottom-up method. The basic idea is that we generate shape samples, and train a color model of the target based on the color data collected according to shape samples in an EM framework. The EM iteration would end up with a color model that maximizes the likelihood of color observations.

At time t, we have $\{(s_t^{s,(n)}, \pi_t^{s,(n)}, \pi_t^{c,(n)}, \pi_t^{(n)})\}$ and a color model M_t. The color model, e.g., a Gaussian mixture model, can be built based on a set of color pixels. The procedure for generating the samples at time $t+1$ is shown in Figure 6.

For each shape sample $s_{t+1}^{s,(n)}$, it is trivial to collect all the color pixels inside the region specified by the shape contour, and estimate its distribution $Z_{t+1}^{(n)}$. The E-step calculate $\pi_{(k)}^{c,(n)} = E(Z_{t+1}^{(n)}, \tilde{M}_{(k)}) = p(Z_{t+1}^{(n)}|\tilde{M}_{(k)})$, the likelihood of such a color pixel distribution given the color model $\tilde{M}_{(k)}$ passed from the previous EM iteration. Every shape sample will be associated with a color likelihood value with respect to a prior color model. After compounding these likelihood probabilities with the priors of the shape samples, the M-step will build a new color model $\tilde{M}_{(k+1)}$ based on these color distributions $Z_{t+1}^{(\cdot)}$ and their beliefs $\pi_{(k)}^{(\cdot)}$ at the $k+1$-th iteration. If a Gaussian mixture model is used, the M-step estimates the parameters of the mixture model based on a set of weighted color pixel values. The EM iteration is expected to converge to a likely color model M_{t+1} that produce the the most likely color observations at time frame $t+1$. The EM iteration in this algorithm basically is a *bottom-up* routine to learn a new color model based on the old one and a set of training data obtained from shape model. It is similar to the *tranductive learning* approach for color tracking in (Wu and Huang, 2000), in which an old color model was transduced to a new model based on the color distributions at current frame.

Generate $\{(s_{t+1}^{s,(n)}, \pi_{t+1}^{s,(n)}, \pi_{t+1}^{c,(n)}, \pi_{t+1}^{(n)}), M_{t+1}\}$ from $\{(s_t^{s,(n)}, \pi_t^{s,(n)}, \pi_t^{c,(n)}, \pi_t^{(n)}), M_t\}$, $n = 1, \ldots, N$:

//Step(1): Shape samples generating
$s_{t+1}^{s,(\cdot)} =$ `I_Sampling(`$\{(s_t^{s,(\cdot)}, \pi_t^{c,(\cdot)})\}$`)`;

//Step(2): Shape observation
$\pi_{(t+1)}^{s,(\cdot)} =$`Shape_Obsrv(`$s_{t+1}^{s,(\cdot)}$`)`;

//Step(3): Collecting of initial color observations
$Z_{t+1}^{(\cdot)} =$`Color_Collect(`$s_{t+1}^{s,(\cdot)}$`)`;
$\tilde{M}_{(0)} = M_t$;

//Step(4): Re-training of color model
`for` $k = 0 : K-1$
    // E-step
    $\pi_{(k)}^{c,(\cdot)} =$`E(`$Z_{t+1}^{(\cdot)}, \tilde{M}_{(k)}$`)`; $\pi_{(k)}^{(\cdot)} = \pi_{(t+1)}^{s,(\cdot)} \pi_{(k)}^{c,(\cdot)}$;

    // M-step
    $\tilde{M}_{(k+1)} =$`M(`$Z_{t+1}^{(\cdot)}, \pi_{(k)}^{(\cdot)}$`)`;
`end`

$M_{t+1} = \tilde{M}_{(K)}$;

Figure 6. Co-inference tracking algorithm II: combining *top-down* and *bottom-up*.

6. Implementation

Section 5 proposed a framework for tracking and integrating multiple cues based on the importance sampling technique. The remainder of this paper will present a specific implementation of a real-time tracker.

6.1. Shape Representation

In our scenarios of face and hand tracking, the targets are roughly round-shaped. Instead of using a detailed shape model such as B-spline models, we employ a conics model to simplify the shape representation. Of course, this conics model is only suitable for certain specific applications, such as tracking human heads or fists. Here, we take a generic form of the conics, i.e.,

$$\mathbf{X}' A \mathbf{X}' + 2B\mathbf{X} + C = 0.$$

A shape template is initialized by conics fitting. The deformation of the shape is governed by an affine transformation,

$$\mathbf{Y} = A\mathbf{X} + \mathbf{t} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \mathbf{X} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}$$

which characterizes the shape space $\mathcal{S}$. Thus, the dimensionality of the shape space $\mathcal{S}$ is 6. Taking the idea of the shape space (Blake and Isard, 1998), we can determine a conic shape given the

template and an affine transformation. The shape samples in our algorithms are drawn in the shape space, i.e., $\mathbf{X}^s = (A_{11}, A_{12}, A_{21}, A_{22}, t_1, t_2)^T$.

6.2. Shape Observation

It is crucial to have an accurate shape observation model in tracking. Our implementation takes a similar approach used in (Blake and Isard, 1998). Edge detection is performed in 1-D along the normal lines of a set of discrete points on a hypothesized contour, shown in Figure 7. Thus, after performing 1-D edge detection along the normal line corresponding to the point on the contour, the image observation of this point reduce to a set of scalar positions $\mathbf{z} = (z_1, \ldots, z_M)$ along the normal line (Blake and Isard, 1998), due to the presence of clutter. The true observation of this contour point $\tilde{z}$ could be any one of these image positions. So,

$$p(\mathbf{z}|x) = qp(\mathbf{z}|\text{clutter}) + \sum_{m=1}^{M} p(\mathbf{z}|x, \tilde{z} = z_m)p(\tilde{z} = z_m)$$

where x is the point on the shape contour and q is the probability that none of such M positions could be taken as an observation.

$$q = 1 - \sum_{m=1}^{M} p(\tilde{z} = z_m).$$

When we assume that any true observation is unbiased and normally distributed with standard deviation σ, and that it is equally likely to observe any of these M positions, i.e., $p(\tilde{z} = z_m) = p$ for all z_m, and that the clutter is a Poisson process with density λ, then,

$$p(\mathbf{z}|x) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma q \lambda} \sum_{m} \exp -\frac{(z_m - x)^2}{2\sigma^2} \tag{10}$$

where λ, q and σ are parameters to model the image observation process. In (Blake and Isard, 1998), a method of learning these parameters was suggested.

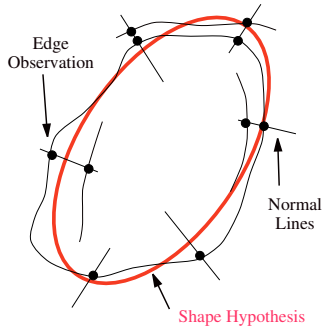

Figure 7. Shape observation and measurement. A set of 1-D observations are applied along the contour of a shape hypothesis. Each 1-D measurement calculates the likelihood of a segment of shape contour by observing some edges.

To measure a shape hypothesis, we can discretize the contour by a set of N contour points, and apply 1-D observations $p(\mathbf{z}_n^s|x_n^s)$, where $n \in \{1, \ldots, N\}$ on these contour points, each of which 1-D edge detection will be performed to detect up to M edge points along the normal line of each contour point. An example is shown in the Figure 7, in which the ellipse (in red) is a shape hypothesis, the curves are the edges detected, and the line segments are the normal lines of the

ellipse. When assuming independent measurement along the contour, we obtain the measurement of a shape hypothesis:

$$p(\mathbf{Z}^s|\mathbf{X}^s) \propto \prod_{n=1}^{N} p(\mathbf{z}_n^s|x_n^s). \tag{11}$$

6.3. Color Representation

We take a parametric color representation in the normalized-RGB color space. If the object is uniform in color, a Gaussian distribution is taken to model the color distribution. For simplicity, we represent the color state by $\mathbf{X}^c = (\mu_{\tilde{r}}, \mu_{\tilde{g}}, \mu_{\tilde{b}}, \sigma_{\tilde{r}}, \sigma_{\tilde{g}}, \sigma_{\tilde{b}})$. If the target has two salient colors, a mixture of two Gaussians is used to model such a distribution. To keep the dimensionality small, we represent the color state by $\mathbf{X}^c = (\mu_{\tilde{r}}^1, \mu_{\tilde{g}}^1, \mu_{\tilde{b}}^1, \mu_{\tilde{r}}^2, \mu_{\tilde{g}}^2, \mu_{\tilde{b}}^2)$. In our experiments, we have used both of these color models. For example, in hand tracking, we use the first one; while in face and head tracking, we employ the mixture model, which is more accurate but needs more computation.

We also experiment with a non-parametric representation by 2D color histogram, which uses two normalized colors such as $\tilde{r}$ and $\tilde{g}$ with N bins. We set $N = 3$ for our approach I and $N = 8$ for approach II. Our experiments show that this non-parametric color model performs poorly in approach I, but works reasonably well in approach II. The examples shown in Section 7 were obtained using the above parametric color representations.

6.4. Color Observation

A set of color pixels is collected inside the shape contour. If the parametric approach is taken, a parametric color model is estimated based on these color pixels, and the Mahalanobis distance is used to measure the similarity of the two distributions.

If non-parametric approach is taken, a color histogram I_s is built based on these color pixels, and the histogram intersection $\phi_c(s)$ (Swain and Ballard, 1991; Birchfield, 1998) is computed between the hypothesis color model $\mathbf{X}^c$ and the observed histogram $\mathbf{Z}^c = I_s$:

$$\phi_c(s) = \frac{\sum_{k=1}^{N} \min(I_s(k), \mathbf{X}^c(k))}{\sum_{k=1}^{N} I_s(k)}. \tag{12}$$

We can use such histogram intersection to approximate the color likelihood, i.e, $p(\mathbf{Z}^c|\mathbf{X}^c) \propto \phi_c(s)$.

7. Experiments

This section reports some experiments of sequential Monte Carlo tracking techniques and our tracking algorithm based on co-inference learning. The tracking performances of both single cue and multiple cues are examined in this section.

7.1. Single Cue

We implement the Condensation algorithm, and run it with two different methods of hypotheses measurements, i.e., shape and color, respectively.

When shape hypotheses are solely measured by edge observations, the algorithm works well in simple backgrounds and where strong edges are observed. However, when the background is cluttered, the tracker usually fails because some samples with high evaluations might be shape distractors. In Figure 8, Red ellipses represent shape samples. The brighter the red ellipse, the higher the evaluation. The blue ellipse shows the shape estimation of the target. In these examples,

many samples have high evaluations on the heavily cluttered backgrounds, such as the keyboard area in Figure 8(a) and the bookshelf area in Figure 8(b).

(a) (b)

Figure 8. CONDENSATION using shape observations alone: many hypotheses are generated on clutter. (a) Tracking hand. The keyboard area produce many false shape hypotheses. (b) Tracking head. The bookshelf area is highly cluttered so that many hypotheses have incorrect shape measurements.

Part of the reason for the failure comes from our conics shape model, which does not accurately align to the contours of the target. Of course, a solution is to use a detailed shape model such as the B-spline model in (Blake and Isard, 1998). It will alleviate the difficulty of clutters because of the uniqueness of the shape representations, which was the direct reason that a leaf could be tracked against a background full of camouflage leafs in (Blake and Isard, 1998). However, it will need more computation. On the other hand, we shall see in the next subsection that adding a rough color model to this rough shape model will make the tracking very robust, while keeping the computation costs low.

(a) (b)

Figure 9. CONDENSATION using color observations alone: color distractors and non-stationary environments made tracking difficult. (a) Tracking face. Many hypotheses were generated for the wooden door area, because many unlikely hypotheses survived the color measurements. (b) Tracking shoulder in a dynamic environment. Due to the changing of the illumination conditions, the color measurements were not accurate anymore.

Another experiment is to evaluate shape hypotheses by making color observations. We construct a color model in the tracking initialization stage. When the hypotheses are solely measured by their color distributions, the tracking algorithm succeeds when the background does not have regions with similar color as the target. However, the tracker usually fails when the background contains color distractors. Figure 9(a) shows the case when the wooden color is similar to the skin tone. In Figure 9(b), the lighting conditions change dramatically, which makes it difficult to track the shoulder of the person.

7.2. MULTIPLE CUES

The co-inference tracking algorithm described in Section 5 has been applied to a variety of environments and tracking tasks. Our experiments show that the tracking algorithm with multiple

cues performs very robustly. The tracking algorithm runs on a 1-processor PIII 850MHz PC at around 30Hz [1].

(a) (b)

Figure 10. The hypotheses produced by the co-inference algorithm. We can see here that by using multiple cues, a set of high quality samples are drawn.

In our algorithm, the distribution of the samples are more concentrated as shown in Figure 10 compared to Figure 8 and Figure 9.

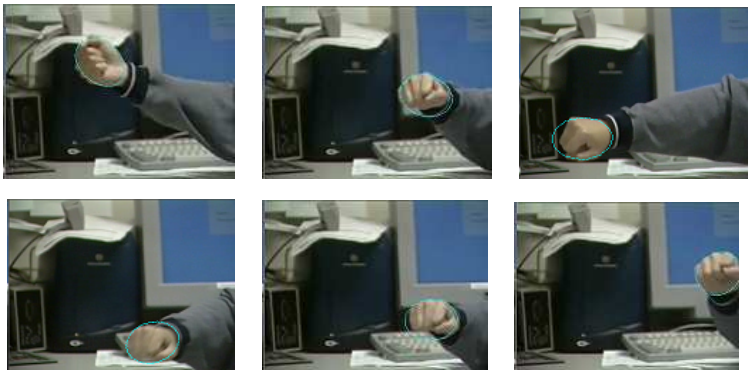

Figure 11. A Hand in clutter.

Figure 11 shows the example of tracking a hand fist under large rotations against a cluttered background. When the tracker is solely based on shape and edge, it loses track when the hand leaves the keyboard area, which has been shown in the previous section. However, our algorithm that integrates both shape and color cues, can overcome this difficulty. The reason behind the success is that different modalities provide reinforcement to each other in a co-inference fashion.

Figure 12 shows the example of tracking a head with out-of-plane rotations in an office environment [2]. Obviously, the color distribution of the girl's head has at least two components, skin color tone and black hair tone. So, when she turns her head around, it displays non-stationary color changes of the visible side of the head. In this scenario, it does not make sense to construct a fixed color model for her head, since the color distribution of the visible side of her head varies when she rotates her head. One of the solutions is to make a 3D head model with color features (Birchfield, 1998). Similar texture model was reported in (Toyama and Wu, 2000). Another approach is to adapt the color model to the lighting changes (Raja et al., 1998; Wu and Huang, 2000). Our co-inference tracking algorithm adapts to the non-stationary color distributions, with a reasonable assumption that the changing of the color distributions is slow and smooth. Our algorithm tracks the head very accurately, even when she moves in front of the wooden door. The reason behind this is that the shape modality provides an external constraint for the color modality.

[1] Some of the demo video sequences, such as `hand.mpg`, `girl.mpg`, `lecture.mpg` and `VE.mpg`, can be obtained from `http://www.ece.northwestern.edu/~yingwu`

[2] The testing sequences were obtained from `http://robotics.stanford.edu/~birch/headtracker`

Figure 12. Face in an office environment. *(Sequence courtesy of Dr. Stan Birchfield.)*

Figure 13. Head in a lecture room with dramatic lighting variations. *(Sequence courtesy of Dr. Kentaro Toyama.)*

Figure 13 shows the case of a lecture room where the lighting changes dramatically due to an overhead projector, and the color of the speaker's head varies in a wide range of intensities. Our algorithm tracks the speaker's head robustly, although it sometimes failed reasonably due to large camera motion and speaker's uncertain movements in very dark lights.

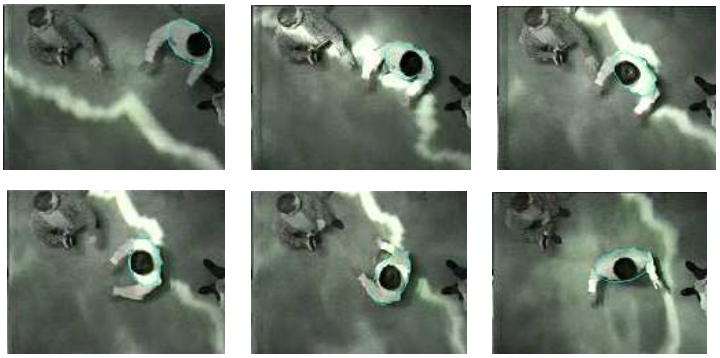

Figure 14. Shoulder in CAVE, a large virtual environment with lighting diffused from screen displays.

Figure 14 shows the tracking scenario in a large virtual environment, which has four displays on three side walls and the floor. The camera is mounted on the ceiling. Of interest is to estimate the user's positions and orientations by tracking his head and shoulder. The difficulty is that the displays will diffuse a large amount of lights to the environments. Tracking the shoulder is even harder than tracking the head, since the shoulder deforms and rotates much more, and as well it does not produce strong edges as the head does. It is a very difficult scenario for the methods

using a single modality. However, employing multiple modalities in the target representation, our algorithm works robustly when parameters, i.e., λ, q and σ mentioned in Section 6, are properly set. In all of our experiments, we adjust parameter settings manually.

Figure 15. A face occluded by another face. *(Sequence courtesy of Dr. Stan Birchfield.)*

Figure 15 shows a good example of occlusions. In this example, another head moves in front of the target, i.e., the girl. The co-inference tracking algorithm tracks the girl when occlusion occurs. The reason behind this is that the occluding object (the boy's head) has a different size from the girl's head, which avoids generating too many hypotheses on the occluding object.

From our extensive experiments on both live videos and recorded sequences, the proposed co-inference tracking algorithm performs very robustly against cluttered backgrounds, non-stationary color changes, slow dynamic illumination environments and some occlusion scenarios.

8. Conclusions

Visual tracking is a fundamental problem in computer vision. It receives more and more attention due to the rapid development of visual surveillance and vision-based human computer interaction applications, in which robust tracking methods are desirable. However, one of the difficulties confronting us is the lack of an effective way for multiple visual cue integration, which prevents accurate and robust methods to evaluate the state hypotheses on the image observations. If richer target representations are used in tracking, more accurate evaluations of the hypotheses can be obtained. Basically, tracking involves a search in the hypothesis space to identify the optimal one that matches the image observations best. Thus, if multiple modalities are used in the target representation, the computation of many tracking methods, such as the CONDENSATION algorithm, will increase dramatically, due to the increase of the dimensionality of the state space.

In this paper we have presented a *co-inference* approach for integrating and tracking multiple cues. The tracking problem is formulated as the inference problem of a graphical model. Our approach is based on the structured variational analysis of a factorized graphical model, which suggests that the inference in a higher dimensional state space can be factorized by several lower dimensional state subspaces (or modalities) in an iterative fashion. We call this *co-inference*. To implement the co-inference for visual tracking, a sequential Monte Carlo tracking algorithm, based on the importance sampling technique, is proposed to simulate and approximate the *co-inference* interactions among different modalities. Our tracking algorithm is robust in dealing with target deformations and color variations, since a richer representation of the target is employed.

The *co-inference* phenomenon is very interesting since it provides an explanation of the information fusion and exchanges among different modalities. We can see that it might be a general

phenomenon for the learning in high dimensional spaces. For example, in the area of text classification, although it does not involve dynamics, an interesting approach, called *co-training* (Blum and Mitchell, 1998), has been developed to explore the correlations among different modalities. Besides the further study of the co-inference itself, we will investigate the occlusion problem and extend our work to the case of tracking multiple objects and articulated objects in the future. In addition, the future investigation will include the study of automatic approaches of estimating the parameters in the co-inference tracking algorithm.

Appendix: The Derivation of the Fixed Point Equations

This appendix presents some details for the variational analysis of the factorized graphical model and the derivation of the fixed point equation described in Section 3. Our purpose is to illustrate the so-called *co-inference* phenomenon in the interactions of different modalities. In the visual tracking scenario, the state variables are continuous, which makes it very difficult to analyze in graphical models. To derive analytical results, we follow the work of Ghahramani and Jordan to make some simplifications here, and investigate the case of discrete states and linear observations, hoping to provide some insights. A similar analysis and another variational inference can be found in (Ghahramani and Jordan, 1997).

To simplify the analysis, we assume the state $\mathbf{X}_t^m$ a multinomial random variable which takes one of K discrete states, i.e., $\mathbf{X}_t^m \in \{1, \ldots, K\}$. Here, we represent this random variable by a $K \times 1$ vector. Only one element of the vector will be 1 while others are 0, which means that the state variable is in one of these discrete states.

$$\mathbf{X}_t^m = \begin{pmatrix} x_{t,1}^m \\ \vdots \\ x_{t,K}^m \end{pmatrix}$$

Such a setting could be used to approximate a continuous case in the tracking scenario. The system dynamics will be approximated by a $K \times K$ transition matrix $P = \{P_{ij}\}$. The inference task of the original factorized model shown in Figure 2 is:

$$P(\underline{X}_t | \underline{Z}_t, \phi) = \frac{1}{Z} exp\{-H(\underline{X}_t, \underline{Z}_t)\}, \tag{13}$$

where ϕ represents the parameters of the graphical model, and Z is a constant to normalize the probabilities, and where

$$\begin{aligned} H(\underline{X}_t, \underline{Z}_t) \; = \; & \frac{1}{2} \sum_{t=1}^{T} (\mathbf{Z}_t - \sum_{m=1}^{M} W^m \mathbf{X}_t^m)' C^{-1} (\mathbf{Z}_t - \sum_{m=1}^{M} W^m \mathbf{X}_t^m) \\ & - \sum_{m=1}^{M} \mathbf{X}_1^{m\prime} \log \pi^m - \sum_{t=2}^{T} \sum_{m=1}^{M} \mathbf{X}_t^{m\prime} (\log P^m) \mathbf{X}_{t-1}^m \end{aligned}$$

where π is the initial probability, and W^m is an observation matrix and C is the covariance matrix, since here we also assume a linear observation model to ease the derivation. The parameters ϕ, including the state transition matrix P, the observation matrix W^m and covariance matrix C, are all known in advance. Such a linear observation assumption may not be true for visual tracking, but we can treat it as an approximation to ease the analysis.

Since the analysis of the original factorized model is difficult, on the other hand, we would write the inference of the structured variational model shown in Figure 3 by:

$$Q(\mathbf{X}_1^m|\theta) = \prod_{k=1}^{K}(h_{1,k}^m \pi_k^m)^{x_{1,k}^m}$$

where $\theta = \{h_{t,k}^m\}$ are the variational parameters, and $\{\pi_k^m\}$ are the initial state probabilities. Since $\mathbf{X}_t^m$ are multinomial random variables, we can write,

$$\begin{aligned} Q(\mathbf{X}_t^m|\mathbf{X}_{t-1}^m, \theta) &= \prod_{k=1}^{K}\left(h_{t,k}^m \sum_{j=1}^{K} P_{kj}^m \mathbf{X}_{t-1,j}^m\right)^{x_{t,k}^m} \quad (14) \\ &= \prod_{k=1}^{K}\left(h_{t,k}^m \sum_{j=1}^{K} (P_{kj}^m)^{x_{t-1,j}^m}\right)^{x_{t,k}^m}. \quad (15) \end{aligned}$$

And similarly, we have,

$$Q(\underline{X}_t|\theta) = \frac{1}{Z_Q} exp\{-H_Q(\underline{X}_t)\}$$

where,

$$H_Q(\underline{X}_t) = -\sum_{m=1}^{M} \mathbf{X}_1^{m\prime} \log \pi^m - \sum_{t=2}^{T}\sum_{m=1}^{M} \mathbf{X}_t^{m\prime} (\log P^m) \mathbf{X}_{t-1}^m - \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbf{X}_t^{m\prime} \log h_t^m.$$

Thus, the KL divergence of $P(\underline{X}_t|\underline{Z}_t, \phi)$ and $Q(\mathbf{X}_1^m|\theta)$ can be written as

$$\begin{aligned} KL(Q||P) &= \langle H\rangle - \langle H_Q\rangle - \log Z_Q + \log Z \\ &= \sum_{t=1}^{T}\sum_{m=1}^{M} \langle \mathbf{X}_t^m\rangle \log h_t^m + \frac{1}{2}\sum_{t=1}^{T}\Bigg[Z_t' C^{-1} Z_t - 2\sum_{m=1}^{M} \mathbf{Z}_t' C^{-1} W^m \langle \mathbf{X}_t^m\rangle \\ &\quad + \sum_{m=1}^{M}\sum_{n\neq m}^{M} tr\left\{W^{m\prime} C^{-1} W^n \langle \mathbf{X}_t^n\rangle \langle \mathbf{X}_t^{m\prime}\rangle\right\} \\ &\quad + \sum_{m=1}^{M} tr\left\{W^{m\prime} C^{-1} W^m diag\langle \mathbf{X}_t^m\rangle\right\}\Bigg] - \log Z_Q + \log Z \end{aligned}$$

where $\langle \mathbf{X}_t^m\rangle = E\left[\mathbf{X}_t^m|\theta, \underline{Z}_t\right]$.

Then, to minimize the KL divergence of such two distributions, we can take the derivative with respect to the structured variational variables h_t^n, equivalently, w.r.t. $\log h_t^n$,

$$\begin{aligned} \frac{\partial KL(Q||P)}{\partial \log h_t^n} &= \langle \mathbf{X}_t^n\rangle + \sum_{t=1}^{T}\sum_{m=1}^{M}\Bigg\{\log h_t^m - W^{m\prime} C^{-1} \mathbf{Z}_t + \sum_{k\neq m}^{M} W^{m\prime} C^{-1} W^k \langle \mathbf{X}_t^k\rangle \\ &\quad + \frac{1}{2}\Delta^m\Bigg\} \frac{\partial \langle \mathbf{X}_t^m\rangle}{\partial \log h_t^n} - \langle \mathbf{X}_t^n\rangle. \end{aligned}$$

where Δ^m is the vector of the diagonal elements of $W^{m\prime} C^{-1} W^m$. Setting the derivatives to zeros, we have

$$\log h_t^m - W^{m\prime} C^{-1} \mathbf{Z}_t + \sum_{k\neq m}^{M} W^{m\prime} C^{-1} W^k \langle \mathbf{X}_t^k\rangle + \frac{1}{2}\Delta^m = 0. \quad (16)$$

Therefore, we end up with a set of fixed point equations:

$$\widetilde{h_t^m} = exp\left\{W^{m\prime}C^{-1}\left[\mathbf{Z}_t - \sum_{k\neq m}^{M} W^k\langle\mathbf{X}_t^k\rangle\right] - \frac{1}{2}\Delta^m\right\}.$$

Since $\langle\mathbf{X}_t^m\rangle = E\left[\mathbf{X}_t^m|\theta, \underline{Z}_t\right]$, the set of fixed point equations can be written as:

$$\widetilde{h_t^m} = exp\left\{W^{m\prime}C^{-1}\left[\mathbf{Z}_t - \sum_{k\neq m}^{M} W^k E\left[\mathbf{X}_t^m|\theta, \underline{Z}_t\right]\right] - \frac{1}{2}\Delta^m\right\}. \tag{17}$$

To see them clearly, we can represent this set of fixed point equations by

$$\widetilde{h_t^m} = \mathcal{G}(\mathbf{Z}_t, \{E[\mathbf{X}_t^n|\underline{Z}_t, \theta] : \forall n \neq m\}) \tag{18}$$

where $\mathcal{G}(\cdot,\cdot)$ is a function. Obviously, this equation is the same as Equation 4 in Section 3.

Acknowledgments

We would like to thank the reviewers for their highly constructive comments and suggestions. This work was supported in part by National Science Foundation Grants CDA-96-24396 and IRI-96-34618, NSF Alliance Program, Northwestern Faculty startup funds, and NSF IIS-03-08222.

References

Azoz, Y., L. Devi, and R. Sharma: 1998, 'Reliable Tracking of Human Arm Dynamics by Multiple Cue Integration and Constraint Fusion'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. Santa Barbara, California, pp. 905–910.

Birchfield, S.: 1998, 'Ellitical Head Tracking Using Intensity Gradient and Color Histograms'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. Santa Barbara, California, pp. 232–237.

Black, M. and A. Jepson: 1996, 'Eigentracking: Robust Matching and Tracking of Articulated Object Using a View-Based Representation'. In: *Proc. European Conf. Computer Vision*, Vol. 1. pp. 343–356.

Blake, A. and M. Isard: 1998, *Active Contours*. London: Springer-Verlag.

Blum, A. and T. Mitchell: 1998, 'Combining Labeled and Unlabeled Data with Co-Training'. In: *Proc. Conf. Computational Learning Theory*. pp. 92–100.

Bregler, C.: 1997, 'Learning and Recognition Human Dynamics in Video Sequences'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. pp. 568–574.

Cham, T.-J. and J. Rehg: 1999, 'A Multiple Hypothesis Approach to Figure Tracking'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2. pp. 239–244.

Comaniciu, D., V. Ramesh, and P. Meer: 2000, 'Real-Time Tracking of Non-Rigid Objects using Mean Shift'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. II. Hilton Head Island, South Carolina, pp. 142–149.

Darrell, T., G. Gordon, M. Harville, and J. Woodfill: 1998, 'Integrated Person Tracking Using Stereo, Color and Pattern Detection'. In: *IEEE Conf. on Computer Vision and Pattern Recognition*. Santa Barbra, pp. 601–609.

Dempster, A. P., N. M. Laird, and D. B. Rubin: 1977, 'Maximum Likelihood from Incomplete Data Via the EM Algorithm'. *J. Royal Statistical Society Series B* **39**, 1–38.

Deutscher, J., A. Blake, and I. Reid: 2000, 'Articulated Body Motion Capture by Annealed Particle Filtering'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. II. Hilton Head Island, South Carolina, pp. 126–133.

Doucet, A., S. J. Godsill, and C. Andrieu: 2000, 'On Sequential Monte Carlo Sampling Methods for Bayesian Filtering'. *Statistics and Computing* **10**, 197–208.

Gavrila, D. M.: 1999, 'The Visual Analysis of Human Movement: A Survey'. *Computer Vision and Image Understanding* **73**, 82–98.

Ghahramani, Z.: 1995, 'Factorial Learning and the EM Algorithm'. In: G. Tesauro, D. Touretzky, and T. Leen (eds.): *Advanced in Neural Information Processing Systems 7*. Cambridge, MA, pp. 617–624, MIT Press.

Ghahramani, Z. and M. Jordan: 1997, 'Factorial Hidden Markov Models'. *Machine Learning* **29**, 245–275.

Hager, G. and P. Belhumeur: 1996, 'Real-time Tracking of Image Regions with Changes in Geoemtry and Illumination'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. pp. 403–410.

Isard, M. and A. Blake: 1996, 'Contour Tracking by Stochastic Propagation of Conditional Density'. In: *Proc. of European Conf. on Computer Vision*. Cambridge, UK, pp. 343–356.

Isard, M. and A. Blake: 1998a, 'CONDENSATION — Conditional Density Propagation for Visual Tracking'. *Int'l Journal of Computer Vision* **29**, 5–28.

Isard, M. and A. Blake: 1998b, 'ICONDENSATION: Unifying Low-level and High-level Tracking in a Stochastic Framework'. In: *Proc. of European Conf. on Computer Vision*, Vol. 1. pp. 767–781.

Jordan, M., Z. Ghahramani, T. Jaakkola, and L. Saul: 2000, 'An Introduction to Variational Methods for Graphical Models'. *Machine Learning* **37**, 183–233.

Li, B. and R. Chellapa: 2000, 'Simultaneous Tracking and Verification via Sequential Posterior Estimation'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. II. Hilton Head Island, South Carolina, pp. 110–117.

Liu, J. and R. Chen: 1998, 'Sequential Monte Carlo Methods for Dynamic Systems'. *J. Amer. Statist. Assoc.* **93**, 1032–1044.

Liu, J., R. Chen, and T. Logvinenko: 2000, 'A Theoretical Framework for Sequential Importance Sampling and Resampling'. In: A. Doucet, N. de Freitas, and N. Gordon (eds.): *Sequential Monte Carlo in Practice*. New York: Springer-Verlag.

MacCormick, J. and A. Blake: 1999, 'A Probabilistic Exclusion Principle for Tracking Multiple Objects'. In: *Proc. IEEE Int'l Conf. on Computer Vision*. Greece, pp. 572–578.

MacCormick, J. and M. Isard: 2000, 'Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking'. In: *Proc. of European Conf. on Computer Vision*, Vol. 2. pp. 3–19.

Pavlović, V., R. Sharma, and T. S. Huang: 1997, 'Visual Interpretation of Hand Gestures for Human Computer Interaction: A Review'. *IEEE Trans. on PAMI* **19**, 677–695.

Rabiner, L.: 1989, 'A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition'. *Proceedings of the IEEE* **77**, 257–286.

Raja, Y., S. McKenna, and S. Gong: 1998, 'Colour Model Selection and Adaptation in Dynamic Scenes'. In: *Proc. of European Conf. on Computer Vision*. pp. 460–475.

Rasmussen, C. and G. Hager: 1998, 'Joint Probabilistic Techniques for Tracking Multi-Part Objects'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. pp. 16–21.

Saul, L. and M. Jordan: 1996, 'Exploiting Tractable Substructures in Intractable Networks'. In: D. Touretzky, M. Mozer, and M. Hasselmo (eds.): *Advances in Neural Information Processing Systems 8*. Cambridge, MA, pp. 486–492, MIT Press.

Swain, M. and D. Ballard: 1991, 'Color Indexing'. *Int'l Journal of Computer Vision* **7**, 11–32.

Tanner, M. A.: 1993, *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*. New York: Springer-Verlag.

Tao, H., H. Sawhney, and R. Kumar: 1999, 'A Sampling Algorithm for Detecting and Tracking Multiple Objects'. In: *Proc. ICCV'99 Workshop on Vision Algorithm*. Corfu, Greece.

Tao, H., H. Sawhney, and R. Kumar: 2000, 'Dynamic Layer Representation with Applications to Tracking'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. 2. pp. 134–141.

Toyama, K. and G. Hager: 1996, 'Incremental Focus of Attention for Robust Visual Tracking'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*. pp. 189–195.

Toyama, K., J. Krumm, B. Brumitt, and B. Meyers: 1999, 'Wallflower: Principles and Practice of Background Maintenance'. In: *Proc. IEEE Int'l Conf. on Computer Vision*. Korfu, Greece, pp. 255–261.

Toyama, K. and Y. Wu: 2000, 'Bootstrap Initialization of Nonparametric Texture Models for Tracking'. In: *Proc. of European Conf. on Computer Vision*. Irland.

Wren, C., A. Azarbayejani, T. Darrel, and A. Pentland: 1997, 'Pfinder: Real-Time Tracking of the Human Body'. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **9**, 780–785.

Wu, Y. and T. S. Huang: 2000, 'Color Tracking by Transductive Learning'. In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Vol. I. Hilton Head Island, South Carolina, pp. 133–138.

Wu, Y. and T. S. Huang: 2001a, 'Hand Modeling, Analysis and Recognition for Vision-based Human Computer Interaction'. *IEEE Signal Processing Magazine* **18**, 51–60.

Wu, Y. and T. S. Huang: 2001b, 'Robust Visual Tracking by Co-inference Learning'. In: *Proc. IEEE Int'l Conference on Computer Vision*, Vol. II. Vancouver, pp. 26–33.