# A Co-inference Approach to Robust Visual Tracking

Ying Wu, Thomas S. Huang
Beckman Institute
University of Illinois at Urbana-Champaign
405 N. Mathews, Urbana, IL 61801
{yingwu,huang}@ifp.uiuc.edu

## Abstract

*Visual tracking could be treated as a parameter estimation problem of target representation based on observations in image sequences. A richer target representation would incur better chances of successful tracking in cluttered and dynamic environments. However, the dimensionality of target's state space also increases making tracking a formidable estimation problem. In this paper, the problem of tracking and integrating multiple cues is formulated in a probabilistic framework and represented by a factorized graphical model. Structured variational analysis of such graphical model factorizes different modalities and suggests a* co-inference *process among these modalities. A sequential Monte Carlo algorithm is proposed to give an efficient approximation of the* co-inference *based on the importance sampling technique. This algorithm is implemented in real-time at around 30Hz. Specifically, tracking both position, shape and color distribution of a target is investigated in this paper. 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 sensor fusion problems.*

## 1 Introduction

Visual tracking is an important problem in visual surveillances and vision-based interfaces. One of the purposes of visual tracking is to infer the *states* of the targets from image sequences. It involves some fundamental research problems such as object representation and matching.

*Bottom-up* and *top-down* approaches are two kinds of methodologies to approach the visual tracking problem. *Bottom-up* approaches generally tend to construct object states by analyzing the content of images. Basically, many segmentation-based methods can be categorized as *bottom-up* approaches. For example, blob tracking techniques group similar image pixels into blobs to estimate the positions and shapes of the target. On the contrary, *top-down* approaches generate candidate hypotheses from previous time frame based on a parametric representation of the target. Tracking is achieved by measuring and verifying these hypotheses against image observations. Many model-based and template-matching methods can be categorized as *top-down* approaches. *Bottom-up* methods could be efficient, yet the robustness is largely limited by the ability of image analysis. On the other hand, *top-down* approaches depend less on image analysis, but their performances are largely determined by hypotheses generating and verification.

Tracking techniques generally have four elements, *target representation*, *observation representation*, *hypotheses generating*, and *hypotheses measurement*, which roughly characterize tracking performances and limitations. To discriminate the target from other objects, *target representation*, including target's geometry, motion, appearance, etc., characterizes the target in a *state space* either explicitly or implicitly. It is a fundamental problem in computer vision. For example, parameterized shapes [12, 13], and color distributions [6, 17, 23] are often employed as target representations. To provide a more constrained description of the target, some methods employ both shape and color [1, 13, 18, 21]. To add uniqueness in the target representation, many methods even employ targets appearance, such as image templates [11, 20] or eigen-space representation [2], as the target representation. Motion could be taken into account in target representations, since different objects can be discriminated by the differences of their motions. 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*. For instance, tracking a person in a crowd would be a challenging visual task. Closely related to *target representation*, *observation representation* defines the image evidence of the object representation. For example, if the target is represented by its contour, we expect to observe edges of the contour in the image. *Hypotheses measurement* evaluates the

0-7695-1143-0/01 $10.00 © 2001 IEEE

matching between hypotheses and image observations. For example, template-matching tracking method often takes SSD as the measurement. The evaluation would be quite challenging when measuring a shape hypothesis in a clutter background. Although some analytical results were reported in [3], many current tracking methods take *ad hoc* measurements. *Hypotheses generating* is to produce new hypotheses based on old estimation of target's representation and old observation. Target's dynamics could be embedded in such a predicting process. Intuitively, hypotheses generating characterizes the search range and confidence level of the tracking. The Kalman filtering technique gives a classical example of hypotheses generating under Gaussian assumptions.

Since image sequences contain very rich visual information, using single object representation would not be robust when the target is in a clutter. A hypothesis of a richer target representation would have better opportunities to be verified according to various aspects of image observations. For example, combining color distribution of the target could largely enhance the robustness of contour tracking in a heavily cluttered background, and integrating shape and color representations could incur better tracking against color distracters. On the other hand, if the target and the clutter are indistinguishable in terms of their representations, the tracking has to be almost determined by the prior knowledge about the dynamics at least in a short period of time. If such prior dynamics is not available or we assume random walk for the target, we could say that the target is *intrackable* in terms of 2D. This aperture problem motivates the research of tracking and integrating multiple visual cues.

Multiple cues integration could be done in terms of object representation and observation representation. Some approaches perform multiple observation measurements, and accumulate the measurements for each hypothesis [1]. Although robust to some extent, many methods of combining the measurements from different sources are often *ad hoc*. To integrate shape and color, many tracking algorithms assume fixed color distribution [13, 21] for the target to enable efficient color segmentation. However, such assumption is often invalid in practice. Instead of assuming fixed color representation, some methods also include color modality in the target representation [4, 18, 22], in which a multivariable Gaussian was used to represent both color and motion parameters. Non-stationary color tracking methods [17, 23] were also reported in the literature. Tracking both shape and color would be a formidable problem, since it increases the dimensionality of the state space of the target. In this scenario, tracking is equivalent to recover the joint states of position, shape and color.

To approach the problem of tracking multiple cues, this paper formulates it as a factorized graphical model. Due to the complexity of such a graphical model, a *variational method* is taken to approximate the Bayesian inference. Different modalities in the model present a *co-inference* phenomenon. Based on the analysis of the factorized model, this paper presents an efficient Monte Carlo tracking algorithm to integrate multiple visual cues, in which the transduction of different modalities is achieved by the EM iterations.

The factorized graphical model will be presented in section 2. Section 3 will describe the techniques in sequential Monte Carlo approaches for tracking problems. Our proposed approach will be presented in section 4, and the details of our tracking implementation and experiments will be described in section 6.

## 2 Graphical Model of Tracking

In a dynamic system, the states of the target and image observations are represented by $\mathbf{X}_t$ and $\mathbf{Z}_t$, respectively. The previous 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 could be formulated as an inference problem with the prior $p(\mathbf{X}_{t+1}|\underline{Z}_t)$, which is a prediction density. We have

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

where $p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1})$ 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 could be represented by graphical models in Figure 1. At time $t$, the obsevation $\mathbf{Z}_t$ 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 Markov property, i.e., $p(\mathbf{X}_t|\underline{X}_{t-1}) = p(\mathbf{X}_t|\mathbf{X}_{t-1})$.

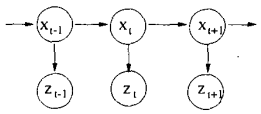


Figure 1: The tracking problem could be represented by a graphical model, similar to the Hidden Markov Model.

The tracking problem can be approached by the inference techniques in graphical model. Consequently, when the dimensionality of the hidden states increases, the inference and learning would become difficult due

to the exponential increase of required computational resources. However, a distributed state representation could largely ease this difficulty by decoupling the dynamics. Fox example, target states could be decomposed into shape states and color states with such the architecture shown in Figure 2(a).

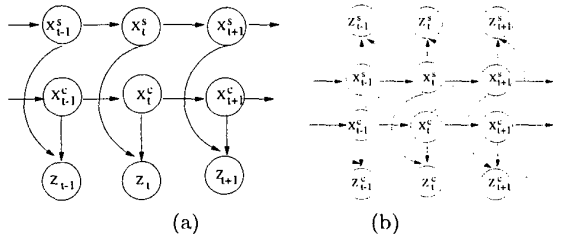

Figure 2: Factorized Graphical Models: (a) The states of the target could be decomposed into shape states $\mathbf{X}_t^s$ and color states $\mathbf{X}_t^c$ in a factorized graphical model. (b) The observation could also be separated into $\mathbf{Z}_t^s$ and $\mathbf{Z}_t^c$.

Furthermore, 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.

Due to the complex structure of the factorized network, the exact inference would be formidable. One approach to this problem is statistical sampling-based methods, such as Gibbs sampling. Another approach is to approximate the posterior probability $p(\underline{X}_t|\underline{Z}_t)$ of the hidden states by a tractable distribution $Q(\underline{X}_t)$. A lower bound on the log likelihood $\log P(\underline{Z}_t)$ can be achieved by such an approximation [10, 14]:

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

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

Generally, we can choose $Q(\cdot)$ to have a simpler structure by eliminating some of the dependences in $P(\cdot)$, while minimizing the Kullback-Leibler divergence between $P(\cdot)$ and $Q(\cdot)$ in equation 2. It can be achieved by a *structured variational inference*. The basic idea is to uncouple the Markov chains and replace the true observation probability of each hidden state by a distinct variational parameter, which can be varied for the minimization. We could write:

$$\begin{aligned} Q(\underline{X}_t|\theta) &= \frac{1}{Z_Q} \prod_{m=1}^{M} Q(\mathbf{X}_1^m|\theta) \prod_{t=2}^{T} Q(\mathbf{X}_t^m|\mathbf{X}_{t-1}^m, \theta) \\ &= \frac{1}{Z_Q} \prod_{m=1}^{M} h_{x_1}^m \pi^m \prod_{t=2}^{T} h_{x_t}^m p(\mathbf{X}_t^m|\mathbf{X}_{t-1}^m) \end{aligned}$$

where $M$ is the number of factorized Markov chains, $\mathbf{X}_t^m$ is the state of the $m$-th modality at time frame $t$, $h_{x_t}^m$ are the variational parameters, and $Z_Q$ is a normalization constant. Although general continuous analysis of such approach is unavailable, [10] presented a structured variational analysis for the case of discrete hidden state and observation. A set of fixed point equations for $h_{x_t}^m$ to minimize $KL(Q||P)$ were obtained [10]:

$$\widetilde{h_{x_t}^m} = g(\mathbf{Z}_t, \{E[\mathbf{X}_t^n|\underline{Z}_t^n, h^n] : \forall n \neq m\}) \quad (3)$$

where $g(\cdot,\cdot)$ is a function that details could be found in the appendix of the paper, $E[\mathbf{X}_t^n|\underline{Z}_t^n, h^m] \equiv \langle \mathbf{X}_t^m \rangle$ is the estimation of the hidden state $\mathbf{X}_t^n$ at the $n$-th uncoupled Markov chain, based on the variational parameters $h^n$. Using these variational parameters, a new set of expectation for the hidden states $\langle \mathbf{X}_t^m \rangle$ will be fed back into equation 3, which can be solved iteratively. It is very similar to the EM algorithm[7]. To make it clear, we could explicitly write up in Equation 4 the fixed point equations in Equation 3 for the case of two modalities, for example, shape and color:

$$\begin{cases} \widetilde{h_{x_t}^s} = g(\mathbf{Z}_t, E[\mathbf{X}_t^c|\underline{Z}_t^c, h^c]) \\ \widetilde{h_{x_t}^c} = g(\mathbf{Z}_t, E[\mathbf{X}_t^s|\underline{Z}_t^s, h^s]) \end{cases} \quad (4)$$

where $\mathbf{X}_t^s$ is the shape state, $\mathbf{X}_t^c$ is the color state, and $h_{x_t}^s$ and $h_{x_t}^c$ represent the shape and color variational parameters, respectively.

It should be noticed that the original densely connected graphical model is uncoupled. The hidden states of each uncoupled Markov chain could be estimated separately, given the set of variational parameters. The estimation of the variational parameters of one of the chains depends on the hidden states of the other chains. If we treat each Markov chain as a modality, this result is quite interesting. We call it *co-inference*, since one modality could be inferred iteratively by other modalities.

The variational analysis of the factorized model in Figure 2 is meaningful for the problem of multiple cues 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 4.

## 3 Monte Carlo Tracking

Sequential Monte Carlo methods for dynamic systems are also studied in the area of statistics [9, 15]. 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 $h(\cdot)$,

$$\lim_{N\to\infty} \frac{\sum_{n=1}^{N} h(s^{(n)})\pi^{(n)}}{\sum_{n=1}^{N} \pi^{(n)}} = E_f(h(\mathbf{X})) \quad (5)$$

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 $\pi^{(n)}$.

Generally, closed-form solutions of dynamic systems are intractable. Monte Carlo methods offer a way to approximate the inference and to characterize the evolution of dynamic systems. Since the posteriori density $p(\mathbf{X}_t|\underline{Z}_t)$ is represented by a set of weighted random samples $\{(s_t^{(n)}, \pi_t^{(n)})\}$, such 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 $t+1$.

When samples are drawn from the prediction prior $p(\mathbf{X}_{t+1}|\underline{Z}_t)$, and sample weights are proportional to the observation likelihood $p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1})$, this sequential Monte Carlo technique is called *factored sampling*, which is an important part of the CONDENSATION algorithm [12]. The robustness of Monte Carlo tracking is due to the maintenance of a pool of hypotheses. Generally, the more the hypotheses, the more chances to get accurate tracking results but the slower the tracking speed. Consequently, the number of samples becomes an important factor in factored sampling, since it determines the tracking accuracy and speed. 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 [5], which retained only the modes (or peaks) of the probability density, and represented the local neighborhood surrounding each mode as a Gaussian distribution. This approach eliminated the need for a large number of samples for representing the distribution around each mode non-parametrically. Different sampling techniques were also investigated to reduce the number of samples. In [16], a partitioned sampling scheme was proposed to track articulated objects. It was basically a hierarchical method to generate the hypotheses. In [8], an annealed particle filtering scheme was taken to search the global maximum of the posteriori probability density.

*Importance sampling* is another Monte Carlo techniques. In practice, it would be difficult to draw random samples from a distribution $f(\mathbf{X})$, samples could be drawn from another distribution $g(\mathbf{X})$, but their weights should be adjusted accordingly. This is the basic idea of *importance sampling*. When samples $s^{(n)}$ are drawn from $g(\mathbf{X})$, but weighted by $\pi^{(n)} = \frac{f(s^{(n)})}{g(s^{(n)})}\tilde{\pi}^{(n)}$, it can be proved that the sample set $\{(s^{(n)}, \pi^{(n)})\}$ is still *properly weighted* with respect to $f(\mathbf{X})$.

To approximate a posterior $p(\mathbf{X}_t|\underline{Z}_t)$, instead of sampling directly from the prior $p(\mathbf{X}_t|\underline{Z}_{t-1})$, samples $s^{(n)}$ could be drawn from another source $g_t(\mathbf{X}_t)$, and the weight of each sample is:

$$\pi_t^{(n)} = \frac{f_t(s_t^{(n)})}{g_t(s_t^{(n)})} p(\mathbf{Z}_t|\mathbf{X}_t = s_t^{(n)}) \tag{6}$$

where $f_t(s_t^{(n)}) = p(\mathbf{X}_t = s_t^{(n)}|\underline{Z}_{t-1})$. 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. The *importance sampling* technique is an important part in the proposed *Co-inference tracking* in section 4.

## 4 Co-inference Tracking

The structured variational analysis of the factorized graphical model in section 2 suggests a way to uncouple the dynamics of the states. In this section, we present an efficient algorithm to approximate the *co-inference* of the variational analysis based on statistical sampling and sequential Monte Carlo technique.

Let $s_t^{(n)} = (s_t^{s,(n)}, s_t^{c,(n)})$ denote the $n$-th sample of the target's state at time $t$, where $s_t^{s,(n)}$ and $s_t^{c,(n)}$ represent shape state and color state of a sample, respectively. $\pi_t^{s,(n)}$, $\pi_t^{c,(n)}$, and $\pi_t^{(n)}$ denote the sample weight based on shape observation, color observation and a combination of shape and color observation, respectively. At time $t$, we have a set of samples associated 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 for 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 3.

The basic idea behind the above iteration is that one modality receives 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 color measurements 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 from shape measurements is difficult to handle clutter backgrounds, especially when a generic shape representation is taken. However, sampling according to color measurements would largely ease this difficulty, since the samples with higher color measurements would have higher probability to propagate. Weight corrections are:

$$\pi_t^{s,(n)} = \frac{f_s(s_t^{s,(n)})}{g_s(s_t^{s,(n)})} p(\mathbf{Z}_t|\mathbf{X}_t = s_t^{(n)})$$

$$f_s(s_t^{s,(n)}) = \sum_{k=1}^{N} \pi_{t-1}^{s,(k)} p(\mathbf{X}_t^s = s_t^{s,(n)}|\mathbf{X}_{t-1}^s = s_{t-1}^{s,(k)})$$

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
$s_{(0)}^{(\cdot)} = s_t^{(\cdot)}$; $\pi_{(0)}^{*,(\cdot)} = \pi_t^{*,(\cdot)}$;

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

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

Figure 3: Co-inference tracking algorithm I: *top-down*

Symmetrically, color samples are then drawn according to shape measurements 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 could 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:

$$\bar{\mathbf{X}}_t^s = \sum_{n=1}^{N} s_t^{s,(n)} \pi_t^{(n)}; \quad \bar{\mathbf{X}}_t^c = \sum_{n=1}^{N} s_t^{c,(n)} \pi_t^{(n)} \qquad (7)$$

Our approach is different from the ICONDENSATION algorithm in [13]. Their method assumes a fixed color distribution and color is used as an extra prior, while our approach could track both shape and color due to the *co-inference* between them. If color dynamics is fixed, our approach would 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 color state could be estimated by taking a bottom-up method. The basic idea is that we generate shape samples but train a color model of the target based 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 observation.

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 procedure for generating the samples at time $t+1$ is shown in Figure 4.

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

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

Figure 4: Co-inference tracking algorithm II: combining *top-down* and *bottom-up*.

The E-step calculates the observation probability for color model hypotheses with respect to the current color model $\tilde{M}_{(k)}$ at different positions and with different shapes. The M-step trains a new color model $\tilde{M}_{(k+1)}$ based on such observations. The EM iteration in the 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 [23].

## 5 Implementation

Section 4 proposed a framework for tracking and integrating multiple cues based on importance sampling technique. The remainder of this paper presents a specific implementation of a real-time tracker.

### 5.1 Shape Representation

Instead of using a detailed shape model by B-splines, we employ conics model for a general purpose, since it is more flexible. The conics model is suitable for certain specific applications, such as tracking human heads or fingertips. 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, $\mathbf{Y} = A\mathbf{X} + \mathbf{t}$, which characterizes the shape space $\mathcal{S}$. Thus, the dimensionality of the shape space $\mathcal{S}$ is 6. A conic shape is determined when given the template and an affine transformation. The shape samples in our algorithms are drawn in the shape space, i.e., $\mathbf{X}^s = (A_{11}, A_{12}, A_{21}, A_{22}, t_1, t_2)$.

### 5.2 Shape Observation

It is crucial to have an accurate shape observation in tracking. Our implementation takes a similar approach used in [3]. Edge detection is performed in 1-D along the normal lines of the hypothesized shapes, shown in Figure 5. Thus, observation reduces to a set of scalar positions $\mathbf{z} = (z_1, \ldots, z_M)$, due to the presence of clutter. The true observation $\tilde{z}$ could be any one of them. 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 $q = 1 - \sum_m P(\tilde{z} = z_m)$. When we assume that any true observation is unbiased and normally distributed with standard deviation $\sigma$, $P(\tilde{z} = z_m) = p$ for all $z_m$, and the clutter is a Poisson process with density $\lambda$, then,

$$p(\mathbf{z}|x) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma q\lambda} \sum_m \exp - \frac{(z_m - x)^2}{2\sigma^2} \quad (8)$$

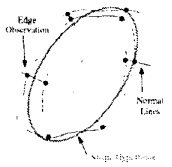


Figure 5: Shape observation and measurement.

### 5.3 Color Representation

We take a parametric color representation in 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 could model such distribution. To keep the dimensionality small, we represent the color state by $\mathbf{X}^c = (\mu^1_{\tilde{r}}, \mu^1_{\tilde{g}}, \mu^1_{\tilde{b}}, \mu^2_{\tilde{r}}, \mu^2_{\tilde{g}}, \mu^2_{\tilde{b}})$.

We also take 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.

### 5.4 Color Observation

A set of color pixels is collected inside the shape contour. If the parametric approach is taken, a parametric color model will be 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 will be built based on these color pixels, and the histogram intersection [1, 19] is computed between the hypothesis color model $\mathbf{X}^c$ and the observed histogram $I_s$:

$$p(\mathbf{X}^c) \sim \phi_c(s) = \frac{\sum_{k=1}^{N} \min(I_s(k), \mathbf{X}^c(k))}{\sum_{k=1}^{N} I_s(k)} \quad (9)$$

## 6 Experiments

The tracking performances of both signal cue and multiple cues are examined in this section.

### 6.1 Single Cue

When the target is solely represented by its shape, a conic in our case, the tracking algorithm works well in simple backgrounds and where strong edges could be observed. However, when the background is cluttered, the tracking often fails because some hypotheses with high probability might be distractors in terms of shape. In Figure 6, many hypotheses have high probability on keyboard (Figure 6(a)) and bookshelf (Figure 6(b)).

(a) (b)

Figure 6: Shape alone: many hypotheses were generated on clutter. (a) Tracking hand, (b) Tracking head

When the target is solely represented by its color distribution, tracking often fails when the background has similar colors to the targets. Figure 7(a) shows the case when the wooden color is similar to skin tone such that false hypotheses are generated. In Figure 7(b), the lighting conditions change dramatically, which makes it difficult to track the shoulder of the person.

(a) (b)

Figure 7: Color alone: color distractors and non-stationary lighting make tracking difficult. (a) Face, (b) Shoulder

### 6.2 Multiple Cues

Our tracking algorithm 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].

Some results are shown in Figure 8. In Figure 8(a), a hand is moving and rotating in a cluttered background. If tracking is solely based on shape and edge, it will be lost when the hand leaves the keyboard area. However, our algorithm, which tracks both color and shape can overcome this difficulty, due to the reinforcement from multiple cues.

Figure 8(b) shows the result of our algorithm to track a head in an office [2]. The subject even turns her head around which makes non-stationary color changes of the visible side of the head. Our algorithm tracks the head very accurately, even when she moves in front of the wooden door.

Figure 8(c) shows the case of a lecture room where the lighting changes dramatically due to an overhead projector. The color of the spearker's head varies in a wide range of intensities. Our algorithm tracks the speaker's head pretty robustly, although it will fail reasonably due to large movements of the camera and speaker's uncertain movements in very dark lights.

Figure 8(d) shows the tracking scenario in a large virtual environment, which has four displays on three sides and floor. The camera is mounted on the ceiling. It is of interest to estimate the user's position and orientation by tracking his head and shoulder. The difficulty is that the displays will diffuse a large amount of lighting in the environments. Tracking the shoulder is even harder than tracking head, since the shoulder deforms more and it does not produce strong edges as the head does. Our algorithm works robustly when parameters are properly set.

Figure 8(e) shows a good example of our algorithm to handle occlusion. The reason behind this example is that the occluding object (the boy) has a different size from the target (the girl), which avoids generating too many hypotheses on the occluding object.

[1]Some demo sequences of our algorithm could be obtained from `http://www.ifp.uiuc.edu/~yingwu`

[2]Thanks Dr. Birchfield for this sequence obtained from `http://robotics.stanford.edu/~birch/headtracker`.

## 7 Conclusions

In this paper we have presented a *co-inference* approach for integrating and tracking multiple cues. This 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 spaces in an iterative fashion. We call this *co-inference*. A sequential Monte Carlo tracking algorithm, based on importance sampling technique, is proposed to approximate the *co-inference* process among different modalities. Our tracking algorithm is robust in dealing with target deformation and color variations, since a richer representation of the target is taken.

The *co-inference* problem is very interesting since it involves the information exchanges between different modalities. We will extend our work to the case of tracking multiple objects and articulated objects in the future.

## Acknowledgments

This work was supported in part by National Science Foundation Grants CDA-96-24396 and IRI-96-34618 and NSF Alliance Program. The authors would like to appreciate the anonymous reviewers for their comments.

## References

[1] Stan Birchfield. Ellitical head tracking using intensity gradient and color histograms. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 232–237, Santa Barbara, California, 1998.

[2] M. Black and A. Jepson. Eigentracking: Robust matching and tracking of articulated object using a view-based representation. In *Proc. European Conf. Computer Vision*, volume 1, pages 343–356, 1996.

[3] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[4] Christoph Bregler. Learning and recognition human dynamics in video sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 568–574, 1997.

[5] Tat-Jen Cham and James Rehg. A multiple hypothesis approach to figure tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 239–244, 1999.

[6] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. Real-time tracking of non-rigid objects using mean shift. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 142–149, Hilton Head Island, South Carolina, 2000.

(a) Hand in clutter.

(b) Face in an office environment.

(c) Head in a lecture room with dramatic lighting variations.

(d) Shoulder in CAVE, a large virtual environment with lighting diffused from screen displays.

(e) A face occluded by another face.

Figure 8: Some results of the Co-inference Tracking algorithm


[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statistical Society Series B*, 39:1–38, 1977.

[8] Jon Deutscher, Andrew Blake, and Ian Reid. Articulated body motion capture by annealed particle filtering. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 126–133, Hilton Head Island, South Carolina, 2000.

[9] Arnaud Doucet, S. J. Godsill, and C. Andrieu. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10:197–208, 2000.

[10] Zoubin Ghahramani and Michael Jordan. Factorial hidden markov models. *Machine Learning*, 29:245–275, 1997.

[11] Greg Hager and Peter Belhumeur. Real-time tracking of image regions with changes in geoemtry and illumination. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 403–410, 1996.

[12] Michael Isard and Andrew Blake. Contour tracking by stochastic propagation of conditional density. In *Proc. of European Conf. on Computer Vision*, pages 343–356, Cambridge, UK, 1996.

[13] Michael Isard and Andrew Blake. ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework. In *Proc. of European Conf. on Computer Vision*, volume 1, pages 767–781, 1998.

[14] Micheal Jordan, Zoubin Ghahramani, Tommi Jaakkola, and Lawrence Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37:183–233, 2000.

[15] Jun Liu and Rong Chen. Sequential monte carlo methods for dynamic systems. *J. Amer. Statist. Assoc.*, 93:1032–1044, 1998.

[16] John MacCormick and Michael Isard. Partitioned sampling, articulated objects, and interface-quality hand tracking. In *Proc. of European Conf. on Computer Vision*, volume 2, pages 3–19, 2000.

[17] Y. Raja, S. McKenna, and S. Gong. Colour model selection and adaptation in dynamic scenes. In *Proc. of European Conf. on Computer Vision*, pages 460–475, 1998.

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

[19] M.J. Swain and D.H. Ballard. Color indexing. *Int. J. Computer Vision*, 7:11–32, 1991.

[20] Hai Tao, Harpreet Sawhney, and Rakesh Kumar. Dynamic layer representation with applications to tracking. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 134–141, 2000.

[21] Kentaro Toyama and Ying Wu. Bootstrap initialization of nonparametric texture models for tracking. In *Proc. of European Conf. on Computer Vision*, Irland, 2000.

[22] C. Wren, A. Azarbayejani, T. Darrel, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 9:780–785, July 1997.

[23] Ying Wu and Thomas S. Huang. Color tracking by transductive learning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume I, pages 133–138, Hilton Head Island, South Carolina, June 2000.