Submitted to **International Conference on Intelligent Robotic Systems 1998**

Joint Probabilistic Techniques for Tracking Objects Using Multiple Visual Cues

Christopher Rasmussen Gregory D. Hager

Department of Computer Science
Yale University
New Haven, CT 06520-8267

Abstract

Autonomous robots relying on vision as a primary sensor frequently must identify and track common objects such as people and cars in order to successfully perform navigation, interaction, and grasping tasks. These objects comprise many visual parts and attributes, yet image-based tracking algorithms are often keyed to only one of a target's identifying characteristics. In this paper, we present a framework for sharing information among disparate state estimation processes operating on the same underlying visual object. Well-known techniques for joint probabilistic data association are adapted to yield increased robustness when multiple trackers attuned to different cues such as color and shape are deployed simultaneously. The utility of each cue varies according to image conditions, necessitating adaptation in the weighting of the various methods to avoid bias. This is achieved by formulating a measure of tracker confidence based on distinctiveness and occlusion probability, which permits deactivating trackers before erroneous state estimates adversely affect the ensemble. We will discuss experiments using color-region-based tracking in tandem with snake tracking that demonstrate the efficacy of this approach.

1 Introduction

More powerful computing hardware and new vision algorithms have expanded the scope of tracking research from its origins in simple geometric shapes to include such complex objects as people and automobiles. For many tasks, techniques for tracking generic edges, curves, blobs, and textures have proven to be applicable with minor modifications to tracking hands, arms, heads, faces, and cars [2, 3, 4, 6].

Despite these advances, most visual tracking algorithms are quite brittle. In particular, many systems are easily confused in commonly occurring visual situations because of their reliance on a single cue or methodology for locating their target. As recent work in multi-cue tracking suggests [12], one way toward robust visual tracking is through exploiting several simultaneously measured visual cues in as flexible a fashion as possible. Approaches to tracking in this spirit have been successful [13], yet even more flexibility may become necessary in order to track increasingly complex objects through a wide range of poses, backgrounds, and lighting conditions.

In this paper we consider some of the issues which arise in constructing vision-based tracking systems that rely on multiple visual cues and part-based decompositions to track complex objects. The probabilistic and joint probabilistic data association filters introduced in [1] serve as a starting point for developing multi-attribute, multi-part tracking methods. We show how object state estimation using an appropriate mixture of color region [7] and snake trackers [11] can be made less sensitive to distraction (clutter) by exploiting inter-part relationships, and also how target occlusion can be accommodated in a natural manner through measures for deciding to "switch" a component tracking algorithm on or off.

2 Using Multiple Cues

A vision-equipped autonomous robot tasked with moving among and interacting with people or other dynamic vehicles must deal with the problem of tracking them as it and they maneuver in a complex visual environment. Robust state estimation processes that furnish information not only about where a target is in robot coordinates, but also about its current posture and orientation, are critical for motion planning and object and gesture recognition modules. The articulation of human bodies and much industrial machinery implies that self-occlusion [9] (where one part of the body moves in front of another) and self-distraction (when similar parts—e.g., hands or grippers—are close to one another) are common hurdles to be overcome. Moreover, in many situations other moving objects and variegated backgrounds can further aggravate problems of occlusion and distraction [6, 1].

Following multiple parts and attributes of an object in parallel can alleviate many such difficulties. Consider a person tracker that regards its target as consisting of two colored regions—a flesh-colored face above a red-colored shirt—and a head silhouette, represented by a snake. The tracker may rely heavily on the red shirt to maintain contact when the person is surrounded by other, distracting faces in a crowded workspace. Using *a priori* knowledge of the geometric relationship between a standing person's torso and head, a rough fix on the image position of the head can be derived from the shirt's image location and scale. If the person walks behind a piece of equipment, leaving only their face visible, the tracker can switch its focus to this part of their body. When the person walks in front of a highly-textured background, the snake may become confused, increasing the tracker's reliance on color cues. If the background is a tan brick wall similar in color to skin, the edge cues used by the snake will be sufficient for disambiguation.

The remainder of this paper will cover the foundations of the tracking skills necessary for the above scenario. We will start with an exposition of the data association background material, describe its extension to include inter-part and inter-attribute constraints, abstract some principles that can be applied to other tracking methods, and discuss our approach to focus-switching, which we term *variable tracker activation*.

3 Data Association Filters

The probabilistic data association filter (PDAF) [1] is an extension of the Kalman filter [1] that casts the problem of data association, or how to update the state when there are multiple measurements and a single target, in a Bayesian framework. The fundamental idea of the PDAF is of the *combined* innovation, computed over the n measurements detected at a given time step as the weighted sum of the individual innovations: $\nu = \sum_{i=1}^{n} \beta_i \nu_i$. Each β_i is the probability of the association *event* θ_i that the ith measurement is target-originated. Also computed is β_0, the probability of the event that none of the measurements is target-originated. These events encompass all possible interpretations of the data, so $\sum_{i=0}^{n} \beta_i = 1$. Details of the calculation of each β_i are given in [1].

The PDAF also develops the notion of a *validation gate*, or an ellipsoidal volume in measurement space, derived from the current estimate and uncertainty of the target state, such that the probability of a target-originated measurement appearing outside of it is negligible. Little accuracy is thus lost by disregarding measurements falling outside the gate. Using a tracking window to limit target search is a common approximation of the validation gate

3.1 Joint PDAF

The distractor model used by the PDAF to calculate each association probability β_i assumes that the target-originated measurement is the only persistent one in the environment. This is a questionable assumption for many distractors, but it certainly does not hold for multi-part objects. Because of the spatial proximity of the parts, one target-originated measurement may often fall within another target's overlapping validation gate. Such persistent interference, were one to simply run a separate PDA filter on each part, could lead to multiple trackers locked onto the same part.

The joint probabilistic data association filter (JPDAF) [1] deals with this problem by sharing information among separate PDAF trackers in order to more accurately calculate association probabilities. The essential result is an exclusion principle of sorts that prevents two trackers from latching onto the same target.

A key notion in the JPDAF is of a *joint event* Θ, or conjunction of possible target-measurement pairings Θ_{jt_j}, where t_j is the index of the target to which measurement j is matched. Because the expression of joint event probabilities is simplified by using the entire surveillance region as each target's validation gate, efficiency is achieved by considering only *feasible* joint events. The first criterion for a feasible joint event is that each measurement comes from exactly one source. We define τ_j to be 0 if measurement j is attributed to noise and 1 if it is associated with a target. Letting δ_t be the number of measurements associated with target t, the second and final feasibility criterion is that $\delta_t \leq 1$.

Let $\omega_{jt}(\Theta) = 1$ if $\Theta_{jt} \subset \Theta$ and 0 otherwise. Then the probability of association between measurement j and target t given measurements Z is $\beta_{jt} = \sum_{\Theta} P(\Theta \,|\, Z)\omega_{jt}(\Theta)$, where:

$$P(\Theta \,|\, Z) = \kappa \prod_{j=1}^{n} [N_j]^{\tau_j} \prod_{t=1}^{T} \gamma_t. \qquad (1)$$

κ contains terms for normalization and scaling, γ_t is a prior probability on the target being visible (see [1] for details), and N_j is the Gaussian PDF $N[\mathbf{z}_j; \hat{\mathbf{z}}^{t_j}, \mathbf{S}^{t_j}]$ for measurement j ($\mathbf{z}_j$ is the measurement value, $\hat{\mathbf{z}}^{t_j}$ is the predicted measurement value for target t_j, and $\mathbf{S}^{t_j}$ is the associated innovation covariance). State estimation is then the same as for the PDAF.

4 Constrained JPDAF: Parts

The JPDAF, originally developed to track aircraft radar returns, does not provide for any constraints on targets to maintain a particular configuration. Such a stipulation could help to distinguish a complex tracked object from the background or other objects. This capability is added by altering the calculation of the probability of a joint event given in (1) to also quantify how well the measurements fit a multi-part object model. We define a *part* as a spatially distinct subtarget physically attached to the object of interest—e.g., hands and a face are parts of the human body.

Intuitively, an object model describes how the likelihood of one part of an object being in a certain state depends on the states of the other parts. A model for an object comprising n parts p_i can be embedded within a probability function $C(\mathbf{Z}, \mathbf{X})$ that quantifies the degree of fit over a given set of feasible matches between the measurements $\mathbf{Z}$ and $\mathbf{X} = \{\mathbf{X}^{t_j}\}$, the matched parts' states. Here we consider the case where C can be decomposed into a product of pairwise constraint functions $C_{ij}(\mathbf{z}_i, \mathbf{z}_j, \mathbf{X}^{t_i}, \mathbf{X}^{t_j})$ (denoted C_{ij}) such that $C(\mathbf{Z}, \mathbf{X}) = \prod_{i=1}^{n} \prod_{j=1}^{n} C_{ij}$. The absence of a constraint between two parts p_i and p_j is indicated by $C_{ij} = 1$. We let $C_{ii} = 1$, and allow $C_{ij} \neq C_{ji}$.

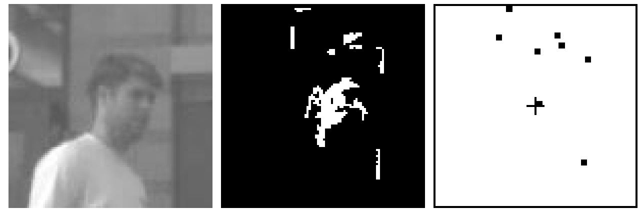
Using inter-part constraints, (1) becomes:

$$P(\Theta \,|\, Z) = \kappa \prod_{j=1}^{n} \left[N_j \prod_{i=1}^{n} [C_{ij}]^{\tau_i} \right]^{\tau_j} \prod_{t=1}^{T} \gamma_t. \qquad (2)$$

For each measurement $\mathbf{z}_j$, the product containing C_{ij} cycles through every other measurement $\mathbf{z}_i$, accumulating how well the relationship between them matches the constraint between their associated targets.

As an example C, consider an object composed of n rigidly linked parts, restricted to translations parallel to the image plane. If measurements for all part trackers are simply image coordinate pairs, then the physical constraints of the system can be captured by a set of image vectors between parts. For each part pair p_i, p_j, an expected measurement difference vector $\mu_{ij} = \mathbf{H}^j \mathbf{X}_0^j - \mathbf{H}^i \mathbf{X}_0^i$ is computed ($\mathbf{H}$ is the measurement matrix from a tracker's filter equations [1]), as well a covariance Σ_{ij} on the expected measurement difference. Then we can define a Gaussian $C_{ij}(\mathbf{z}_i, \mathbf{z}_j, \mathbf{X}^{t_i}, \mathbf{X}^{t_j}) = N[\mathbf{z}_j - \mathbf{z}_i; \mu_{t_i t_j}, \Sigma_{t_i t_j}]$.

The approach to an object model here should be distinguished from other methods that affect state update directly [10, 11]. This formulation simply biases a probabilistic state estimator to favor, when there is a choice, an interpretation of the data that best matches the target model.



(a) (b) (c)

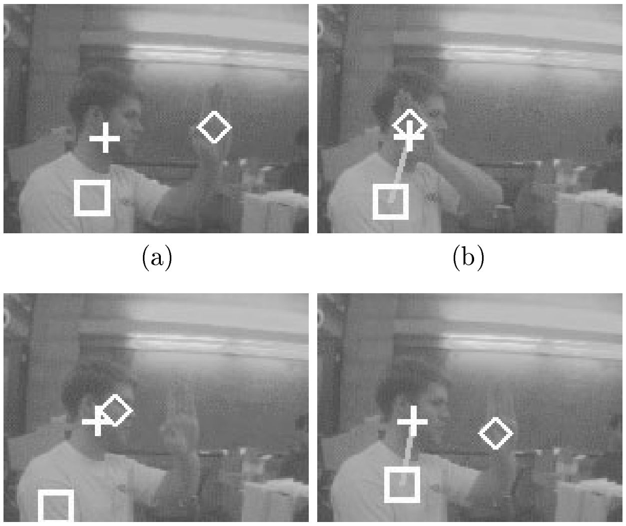
Figure 1: Color Regions and PDAF (from an MPEG). (a) Tracking window; (b) Largest connected components of flesh color; (c) Measurements derived from their centroids.

4.1 Color Regions as Parts

We now discuss applying the PDAF and JPDAF to parts consisting of uniformly colored regions [7]. A part's color is formally defined by pixel membership in a five-dimensional ellipsoid in image-RGB space with mean and covariance $[\mu, \Sigma]$. For reasons explained in [7], the state $\mathbf{X}$ of a color part is restricted to the ellipsoid mean $\mu = [x, y, r, g, b]^T$, while Σ is retained as a fixed parameter. The state is initialized by computing the principal components of manually-sampled target pixels. We have found that a static dynamical model with relatively high process noise often works well for tracking people's body parts.

For the filter update at time t, measurements are first validated by eliminating image pixels outside the ellipsoid $[\mu_t, \Sigma]$ (a rectangular tracking window serves as the image-spatial gate). To facilitate computation of the association probabilities β, the remaining pixels must be converted to point-like measurements. Each pixel could be a separate measurement, but this would be combinatorially cumbersome and it loses the concept of a region. Instead, the mean positions and colors of the largest connected components (CC) of the validated pixels are used as measurements. This approximation gives good results as long as each CC is relatively compact. The process is illustrated in Figure 1.

Application of the PDA and JPDA filters is straightforward after the completion of these steps. Our implementation of a constrained JPDAF tracker uses the same C_{ij} as the example in the previous section, except that measurements have an additional $[r, g, b]^T$ color component, increasing the dimensionality of the Gaussian. This constraint model is an adequate description of the situation when tracking a person's face, shirt, and pants while sitting or walking. Just one rigid constraint is often sufficient to discriminate an object in an otherwise distracting situation. Parts attached in a non-rigid way, such as hands, can be in-



(a) (b) (c) (d)

Figure 2: Avoiding distraction (from an MPEG). (a) Initial states of color parts; (b) Face and hand overlap (from constrained JPDAF sequence); (c) PDAF: hand tracker sticks to face; (d) JPDAF with constraint between face and shirt: trackers stay with correct targets.

corporated by specifying no constraints between them and other parts.

Figure 2 shows the utility of the JPDAF and constrained JPDAF for avoiding mistracking. Color-based trackers are initialized on a person's hand, face, and shirt; the hand then passes in front of the face and moves away. When running as independent PDAF's the hand tracker, attracted to skin color, often "sticks" to the face after the hand is removed, and vice versa. Using an MPEG for identical image conditions and initialization, we confirmed that a JPDAF avoids this problem because the eventuality of the hand and face tracker locking onto the same color region is excluded as an infeasible joint event. However, the JPDAF does not prevent the hand and face trackers from switching places when their paths cross. A constraint to prefer candidate face regions at a vertical offset from the current shirt tracker state effectively anchors the face tracker to the shirt. Although the hand cannot be distinguished from the face while they are overlapping, when it is moved away from the expected face position it is disfavored.

5 Heterogeneous Cues: Attributes

Thus far we have limited our discussion of tracking objects using multiple cues to tracking them according to their parts. Each part tracker has used a single method—color so far, but one could easily imagine a collection of snakes, as with [10], in which coupled snakes track a person's mouth and head (using a different approach to constraints). In this situation, the JPDAF guards against multiple trackers "clumping" on the same target or swapping places when their respective targets are brought into proximity. We now discuss another kind of cue, which we call an *attribute*, not susceptible to this phenomenon.

An attribute is a visual characteristic such as color, edges, texture, depth, or motion. Fundamentally, a part is *what* a tracker tracks, while an attribute is *how* the tracker identifies its target. By its nature, a single part can possess multiple attributes, so it does not make sense to retain a JPDAF-style exclusion principle that prevents multiple trackers of different modalities from following the same target. However, constraints do apply: a color region tracker and a B-spline snake [2] both locked onto a hand, for instance, could be expected to have coincident centers of image mass, or the angle of the major axis of the region could be expected to agree with that of the B-spline.

Different kinds of trackers have distinct measurement spaces, so a separate JPDAF tracker must be run for each attribute. Suppose an object is tracked with m attributes, where each attribute has n_b parts. Then let C_{ij}^{ab} be the constraint function between the ith part of the ath attribute and the jth part of the bth attribute. $C_{ij}^{aa} = C_{ij}$, the familiar single JPDAF inter-part constraint. We modify (2) as follows for the JPDAF on attribute a:

$$P_a(\Theta \,|\, Z) = \kappa \prod_{j=1}^{n} \left[N_j \prod_{b=1}^{m} \prod_{i=1}^{n_b} [C_{ij}^{ab}]^{\tau_i^b} \right]^{\tau_j^a} \prod_{t=1}^{T} \gamma_t. \quad (3)$$

The superscript on each τ is to clarify which attribute generated the measurement, since there are m different sets of measurements. Note that this formula reduces to (2) when there is only one attribute ($m = 1$).

(3) can be directly applied to tracking objects with attributes as amenable to a point-like characterization as color regions, such as motion blobs and disparity maps. Combining snakes with regions seems like a more challenging and fruitful endeavor, but for a number of reasons it is problematic to cast snakes in a JPDAF mold. Kalman snakes [11], for example, effectively use nearest-neighbor data association [1] with short lines as validation gates; considering multiple edges within search windows large enough that they all overlap would lead to a combinatorial explosion. (See also [4] for more discussion of this). Nonetheless, we believe that the JPDAF exclusion principle and

the constraint function machinery that we have developed here can be abstracted and applied to other state estimation techniques. In particular, we have been investigating modifications to the Condensation algorithm [4, 5] for state estimation.

5.1 Condensation: Color and Snakes

The Condensation algorithm has been used, with more complexity than we can do justice to here, to successfully track snakes in highly cluttered environments. Very briefly, the algorithm stochastically estimates state by maintaining a pool of n sample states for each object tracked. Every iteration, the fitness of each sample state $\mathbf{s}_i$ is measured against the current image $\mathbf{Z}$ as a conditional probability $P(\mathbf{Z} \mid \mathbf{s}_i)$. The $P(\mathbf{Z} \mid \mathbf{s}_i)$'s are then normalized to sum to 1. Using a probabilistic scheme tied to the object dynamics, more sample states are generated in the state space neighborhoods of fit states, and less fit states are removed from the pool, always maintaining a constant n. One can estimate the object state as $\hat{\mathbf{X}} = \sum_{i=1}^{n} \mathbf{s}_i P(\mathbf{Z} \mid \mathbf{s}_i)$.

We have implemented a simple version of a Condensation snake tracker, as well as a Condensation analog of the color region tracking method described in Section 4.1 [8], in order to demonstrate some of the benefits of multi-attribute tracking. Figure 3 illustrates how more than one of an object's attributes may be necessary to distinguish it. In the image are four objects, each with a different shape and color attribute: the first row is black, the second green; the first column is egg-shaped, the second is gourd-shaped. Superimposed on the image are the sample states ($n = 50$) of a snake Condensation tracker running on the green egg and a color Condensation tracker running on the black gourd. When the tracked objects are moved near another object with a similar value for the attribute being tracked—here the black egg—distraction results. This difficulty cannot be overcome by adding more parts to the object model, as before, but only by adding an attribute. By considering color and shape jointly, the tracked object becomes unique in this image environment.

To implement a constrained Condensation tracker, we adapt (3) by modifying the calculation of $P_{p_i}(\mathbf{Z} \mid \mathbf{s})$ for each part p_i's sample states to incorporate $\hat{\mathbf{X}}_{p_j}$, the estimated states of the other parts, via a modified constraint function C_{ij} for each pair of parts. More details and preliminary results are given in [8].

6 Variable Tracker Activation

Tracking *failure* [12] occurs when contact with a target is lost, either from occlusion or because clutter distracts the tracker away from the true target. The

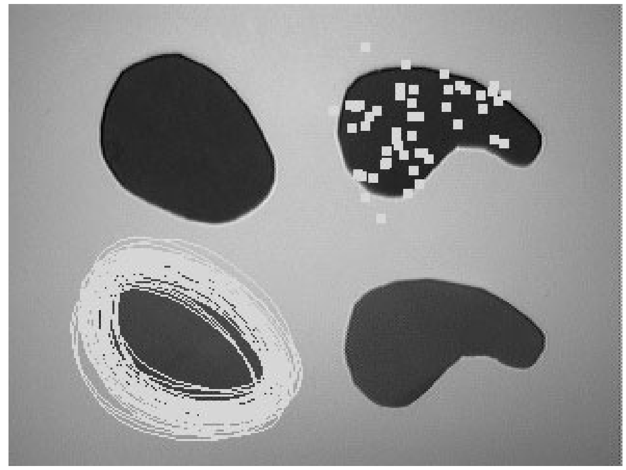


Figure 3: A group of objects unique only when conjunctions of their attributes are considered together.

JPDAF and constrained JPDAF try to prevent failure due to distraction, but they can not completely eliminate it. When tracking an object with multiple parts or attributes, it is desirable to cease state estimation temporarily for any failing parts. Otherwise, erroneous state estimates may be propagated from failing trackers to healthy ones linked by constraints.

Our framework for recognizing and managing partial failure follows from a notion of tracker *confidence*, or a tracker's self-estimated probability of mistracking based on image conditions and its capabilities. Many phenomena bear on confidence, but here we limit ourselves to two parameters that can be estimated heuristically fairly easily. We quantify the confidence χ of the tracker of a part p_i as a combination of the estimated probabilities of part occlusion $P(O_{p_i})$ and distraction $P(D_{p_i})$, the latter of which can be characterized as a function of part distinctiveness. Confidence is only as high as the greatest source of uncertainty allows, so $\chi(p_i) = 1 - \max(P(O_{p_i}), P(D_{p_i}))$. Details of the calculation of $P(O_{p_i})$ and $P(D_{p_i})$ under the constrained JPDAF framework are given below. We are currently investigating heuristics for these values for Condensation tracking.

When $\chi(p_i)$ of the tracker for a part p_i with constraint links at least one other part falls below some threshold $\rho \in [0, 1]$, it is *deactivated*. This means that its image-based state estimation is switched off discretely, and $C_{ij} = 1$ for purposes of calculating (2). The state of p_i is instead chosen to maximize $\prod_{j=1}^{n} C(\mathbf{H}^j \mathbf{X}^j, \mathbf{H}^i \mathbf{X}^i, \mathbf{X}^j, \mathbf{X}^i)$ (for the rigid object described in Section 4, this works out to the average of the states of those parts linked to p_i plus their respec-

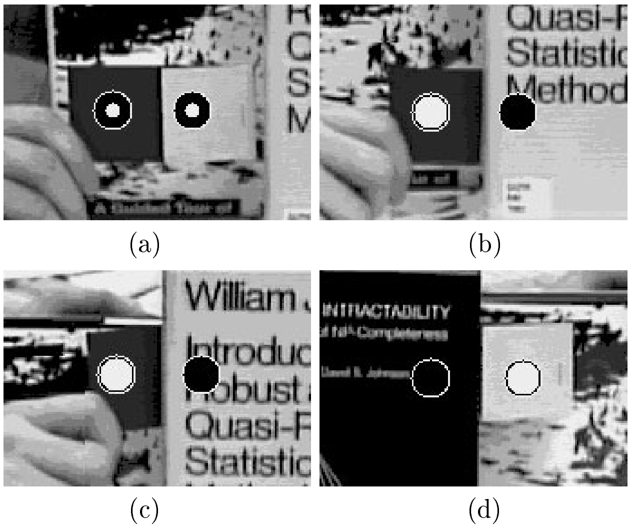


Figure 4: Tracker activation/deactivation due to occlusion. (a) Parts p_1, p_2 unoccluded, active; (b) Part p_2 occluded, inactive; (c) Inactive part p_2's state passively estimated from active part p_1 as object moves; (d) Part p_1 becomes occluded, inactive as part p_2 is reactivated.

tive state difference vectors). While a tracker is deactivated it continues to perform confidence estimation until $\chi(p_i) \geq \rho$, at which point it is re-*activated*.

The probability of a target t being occluded can be derived directly from the JPDA filter as $P(O_t) = \beta_{0t}$, the probability that none of the observed measurements are associated with the target. We estimate the probability of a tracker being distracted by measuring the distinctiveness of its target in the image environment. Viewing the data association problem as one of cluster assignment, distinctiveness is intuitively a measure of cluster separation. With the JPDAF, a target t's distinctiveness depends on how much more probable one association is than the rest. The peakedness of the association probabilities β_{jt} can be captured to a large extent by their entropy:

$$H_t = -\sum_{j=0}^{n} \beta_{jt} \log \beta_{jt} \tag{4}$$

When $\beta_{jt} = 1$ for some j, entropy is minimized because there is no uncertainty about which association to make. When all βs are equal, the entropy is maximal for a particular number of measurements n, meaning that no association is more likely than any other. We convert the entropy of a target t to a probability of distraction with the formula $P(D_t) = H_t/H_{max}$, where H_{max} is the maximal entropy for the current number of measurements.

The tracker activation framework is demonstrated in Figure 4 for a two-part object with a rigid constraint. Despite occlusions first of one part and then the other, tracking proceeds smoothly through multiple activation switches as the state of the currently occluded part is derived from its unoccluded partner or the image as conditions warrant. A white inside black circle indicates normal state estimation, a black circle indicates passive estimation via a constraint, and a white circle indicates normal state estimation while driving another part's passive estimation.

7 Conclusion

We have found that defining a target as a conjunction of parts and attributes and using an intelligent focus-switching scheme lessens the chances of distraction and susceptibility to occlusion, enabling state estimation to proceed in a broader range of visual situations. Work continues on extending these techniques to more tracking algorithms and object models.

References

[1] Y. Bar-Shalom and T. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

[2] A. Blake, M. Isard, and D. Reynard. Learning to track the visual motion of contours. *Artificial Intelligence*, No. 78, pp. 101-133, 1995.

[3] G. Hager and P. Belhumeur. Real-Time Tracking of Image Regions with Changes in Geometry and Illumination. In *CVPR '96*, pp. 403-410, 1996.

[4] M. Isard and A. Blake. Contour Tracking by Stochastic Propagation of Conditional Density. In *ECCV '96*, pp. 343-356, 1996.

[5] M. Isard and A. Blake. A Mixed-state Condensation Tracker with Automatic Model-switching. In *ICCV '98*, pp. 107-112, 1998.

[6] D. Koller, J. Weber, and J. Malik. Robust Multiple Car Tracking with Occlusion Reasoning. In *ECCV '94*, pp. 189-196, 1994.

[7] C. Rasmussen and G. Hager. An Adaptive Model for Tracking Objects by Color Alone. Techical Report, DCS-TR-1200, Yale University, 1997.

[8] C. Rasmussen and G. Hager. Color Region Tracking Using the Condensation Algorithm. Techical Report, DCS-TR-1222, Yale University, 1998.

[9] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In *ICCV '95*, pp. 612-617, 1995.

[10] D. Reynard, A. Wildenberg, A. Blake, and J. Marchant. Learning Dynamics of Complex Motions from Image Sequences. In *ECCV '96*, pp. 357-368, 1996.

[11] D. Terzopoulos and R. Szeliski. Tracking with Kalman Snakes. In *Active Vision*, A. Blake and A. Yuille, eds., pp. 3-20, 1992.

[12] K. Toyama and G. Hager. Incremental Focus of Attention for Robust Visual Tracking. In *CVPR '96*, pp. 189-195, 1996.

[13] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-Time Tracking of the Human Body. In *SPIE*, Vol. 2615, 1995.