On-line Simultaneous Learning and Tracking of Visual Feature Graphs

Arnaud Declercq and Justus H. Piater
Montefiore Institute (B28), University of Liège
B-4000 Liège, Belgium
{Arnaud.Declercq,Justus.Piater}@ULg.ac.be

Abstract

Model learning and tracking are two important topics in computer vision. While there are many applications where one of them is used to support the other, there are currently only few where both aid each other simultaneously. In this work, we seek to incrementally learn a graphical model from tracking and to simultaneously use whatever has been learned to improve the tracking in the next frames. The main problem encountered in this situation is that the current intermediate model may be inconsistent with future observations, creating a bias in the tracking results. We propose an uncertain model *that explicitly accounts for such uncertainties by representing relations by an appropriately weighted sum of informative (parametric) and uninformative (uniform) components. The method is completely unsupervised and operates in real time.*

1. Introduction

Graphical models are popularly used to represent objects in terms of local appearance and spatial relations for detection, classification and tracking applications [3, 12, 14, 15]. Unsupervised learning of such feature graphs from images is difficult because of the correspondence problem of features between images. If learning is based on a video sequence, tracking can be used to solve this problem. On the other hand, tracking a set of features requires a good model if we want to avoid problems due to occlusion and clutter. In this paper, we address these two tasks simultaneously in a way that they help each other.

The idea is to incrementally learn models of relations that exist between local features, and to use what we have already learned to improve tracking. Since tracking is on-line, it is important to use models with a low computational cost. In practice, we are only interested in learning specific types of relations (rigid, articulated, ...). For this purpose, parametric models or mixtures of a small number of Gaussians are more suitable than nonparametric models.

In this paper, we seek to identify, while tracking, approximately rigid spatial relations whose distributions are roughly Gaussian-shaped, and simultaneously use them to aid the tracking by biasing the observation likelihoods. This will be helpful if the learned model is accurate, but detrimental if it is not.

The three main sources of undue bias are observations that are far from Gaussian, inaccurate parameter estimates based on few observations early during learning, and parameters that change over time. For example, consider an object sitting on a table: The system will learn rigid relations between the object and the table that have to be unlearned when the object is moved.

To manage such situations, we develop in this paper an *uncertain graphical model* that explicitly accounts for its predictive power by representing each pairwise potential as a mixture of a Gaussian (reliable relation) and a uniform density (ignorance).

After discussing some background in Section 2, the uncertain model is introduced in Section 3. The method for tracking the features using the current, learned model is explained in Section 4. Section 5 describes the learning of an uncertain model. Finally, experimental results are presented in Section 6.

2. Related Work

Appearance- and structure-based approaches have been widely used for object representation. Appearance-based models usually model an object as color histograms [2], image templates [1] or neighborhood features such as SIFT. Structural models typically represent an object using 2D edges or 3D geometric models [7]. A natural way to combine the shape and appearance of an object is to use a graphical model whose nodes and edges correspond to local features and spatial relations between them [13, 14, 15].

On-line tracking is often addressed by using a prior model of the object. This model can be learned from training examples during a learning phase [13]. Another solution is to design the prior model of the object by hand [14]. This means that the model is specific to a particular object and is difficult to reuse in other situations. The model can also

be defined in starting frames [1, 10], taking the risk that it no longer corresponds to changed appearances of the object over time. If we are only concerned with tracking, we can also use a dynamic model that is updated to accommodate the object appearance and structural changes [15].

Unsupervised learning of object models is computationally demanding because it has to find feature correspondence between images [12]. The temporal information from video can be used to solve this problem. Leordeanu and Collins [8] use tracking to group features into objects by observing their co-occurrences. Ramanan *et al*. [11] use a video sequence to build models of animals from temporally coherent clusters that represent body parts. While the former work does not use the learned relationships between parts to refine the matching process, the latter does not allow corrections of the model once it as been learned.

Some recent approaches have been developed to simultaneously learn and track object models. Lim *et al*. [9] propose a method that incrementally learns and adapts a low-dimensional eigenspace representation to reflect appearance changes of the target, thereby facilitating the tracking task. The same kind of model is used by Dowson and Bowden [4] by their N-tiers model that represents both the appearance and the structure of the object. These models do not express the uncertainty inherent in the current model. Moreover, they suppose that the selected region of interest only contains the object to learn.

3. The Uncertain Model

In this paper, we develop an *uncertain model* of relations between features that explicitly accounts for their uncertainty. Thus, a relation will contribute stability to tracking without exerting overly strong bias that would hamper tracking. For now, features and relations are initialized manually; in principle, one might use e.g. any suitable keypoint detector and triangulation schemes. No assumptions are made regarding the number of objects in a video; in fact, features are not explicitly grouped into objects at all. More or less informative (Gaussian-like) relations are learned between features; coherently moving, rigid objects might then be identified as rigid subgraphs.

To model the features and their relations, we use an undirected graph where the nodes and edges represent the visual features and the relations between them, respectively. We denote the state of each feature at time t by $x_{i,t}$ and its associated image observation by $z_{i,t}$. The joint target states and the joint observations are respectively denoted by $X_t = \{x_{1,t}, \ldots, x_{N,t}\}$ and by $Z_t = \{z_{1,t}, \ldots, z_{N,t}\}$. We also denote $Z_{0:t} = \{Z_1, \ldots, Z_t\}$, the joint observations of the whole sequence until time t.

A spatial relation between two features is represented by a potential function $\psi_{i,j,t}(x_{i,t}, x_{j,t})$ that expresses the constraint on the relative position of the features i and j. This pairwise potential depends on time t because it is learned on-line from the observations up to time $t-1$.

For a Markov network, the probability of the posterior of the joint state X_t given the image measurements Z_t can be written [5]

$$P(X_t|Z_t) = \frac{1}{Z_Q} \prod_{(i,j)\in E} \psi_{i,j,t}(x_{i,t}, x_{j,t}) \prod_{i\in V} p_i(z_{i,t}|x_{i,t}), \tag{1}$$

where Z_Q is a normalization constant, E is the set of edges in the graph, and V is the set of nodes.

Any suitable feature detector and associated observation model $p(z|x)$ can be used. In the following, we focus on the calculation of the potential functions for all edges in E. These functions represent not only an estimate of the relations between the features but also a corresponding measure of uncertainty. We distinguish two sources of uncertainty:

Uncertainty due to tracking. During incremental learning, observed relations may not come from a stationary distribution. For example, a feature may remain static up to time $t-1$ but start moving at time t. In this case, the learned model is no longer predictive of future observations, and its uncertainty is increased.

Uncertainty due to learning. The distribution of observations may correspond to a lesser or greater extent to the parametric model of interest, giving rise to higher or lower predictive uncertainty. Moreover, the uncertainty of a learned model decreases with the number of observations; even observations drawn from a parametric distribution of interest are of little predictive value at early stages of learning.

The following sections detail the representation and estimation of these uncertainties.

4. Tracking Features Using Uncertain Potential Functions

The above Markov network is a generative model at one time instant. To track it, we extend Eqn. 1 to account for a time dimension. Under the conventional Markov assumption for the time dimension,

$$P(X_t|X_{t-1}) = \prod_i p(x_{i,t}|x_{i,t-1}), \tag{2}$$

the posterior probability of the joint state X_t given the image measurements $Z_{0:t}$ can be expressed as

$$\begin{aligned} P(X_t|Z_{0:t}) = \frac{1}{Z_Q} \prod_{(i,j)\in E} \psi_{i,j,t}(x_{i,t}, x_{j,t}) \\ \times \prod_{i\in V} p_i(z_{i,t}|x_{i,t}) p(x_{i,t}|z_{i,0:t-1}), \end{aligned} \tag{3}$$

where

$$p(x_{i,t}|z_{i,0:t-1}) = \int p(x_{i,t}|x_{i,t-1})p(x_{i,t-1}|z_{i,0:t-1})\mathrm{d}x_{i,t-1}. \tag{4}$$

4.1. Sequential Belief Propagation with Known Potential Functions

Let us first consider the problem with known potential functions, i.e., the relations learned until time $t-1$ correctly predict the relations at time t. As shown by Hua and Wu [6], inference can be performed by Sequential Belief Propagation through an iterative, local message passing process. The local message passed from node i to node j at time t and iteration n is given by

$$\begin{aligned} m^n_{j,i,t}(x_{j,t}) \leftarrow \int_{x_{i,t}} \Big[& p_i(z_{i,t}|x_{i,t})\psi_{i,j,t}(x_{i,t},x_{j,t}) \\ & \times \int_{x_{i,t-1}} p(x_{i,t}|x_{i,t-1})P(x_{i,t-1}|Z_{0:t-1})\mathrm{d}x_{i,t-1} \\ & \times \prod_{k\in N(x_{i,t})\setminus j} m^{n-1}_{i,k,t}(x_{i,t}) \Big] \mathrm{d}x_{i,t}, \end{aligned} \tag{5}$$

and the marginal posterior probability at time t is given by

$$\begin{aligned} P^n(x_{i,t}|Z_{0:t}) \propto\ & p_i(z_{i,t}|x_{i,t}) \prod_{j\in N(x_{i,t})} m^n_{i,j,t}(x_{i,t}) \\ & \times \int_{x_{i,t-1}} p(x_{i,t}|x_{i,t-1})P(x_{i,t-1}|Z_{0:t-1})\mathrm{d}x_{i,t-1}. \end{aligned} \tag{6}$$

4.2. Sequential Belief Propagation with Uncertain Potential Functions

In our scenario, the potential functions are learned incrementally, which sometimes leads to situations where an observation at the current time t is in fact not well predicted by the relations learned up to time $t-1$. This may happen if, for example, the two features connected through a relation were motionless until time $t-1$, and one of them starts to move at time t. In this case, it is clear that the rigid relation learned from previous frames is no longer appropriate to track these features. To account for this *uncertainty due to tracking*, we need to augment accordingly the variance of the learned relations. Therefore, in the following we distinguish between the potential $\psi^+_{i,j,t-1}$ learned up to and including time $t-1$, and its variance-augmented counterpart $\psi^-_{i,j,t}$ that replaces $\psi_{i,j,t}$ in Eqn. 5.

A given relation is only used for tracking in the single next frame. Given that the observations in a video are spatially correlated over time, the next observations will not be too far from the current models. If we assume that the (application-dependent and fixed) likelihood of making an observation a distance Δ away from the learned model follows a Gaussian distribution of variance σ^2_Δ, a suitably augmented potential $\psi^-_{i,j,t}$ can be obtained by convolving the learned model $\psi^+_{i,j,t-1}$ with a zero-mean Gaussian with a variance of this order of magnitude:

$$\psi^-_{i,j,t} = \psi^+_{i,j,t-1} \circledast N(0,\sigma_\Delta) \tag{7}$$

The potential functions $\psi^+_{i,j,t-1}$ are learned from the previous frames and account for the uncertainty due to learning. We will see in the next section how they are calculated.

5. Learning the Uncertain Potential Functions

In this paper, we are interested in learning relations of Gaussian shape. Therefore, the informative part of our potential functions is of the form

$$\psi_{i,j,t}(x_{i,t},x_{j,t}) = \mathrm{e}^{-\frac{(r_t-\mu_t)^2}{2\sigma_t^2}}, \tag{8}$$

where r_t corresponds to the observation of the relative position $x_{i,t}-x_{j,t}$ between two given features i and j at time t. The parameters μ_t and σ^2_t are the value of the estimated rigid relation and the variance of the observations around this position.

After a brief review of the incremental estimation of Gaussian parameters, we introduce a potential function for the uncertain model $\psi^+_{i,j,t}(x_{i,t},x_{j,t})$ that augments the observed variance and adds a uniform term to account for two sources of uncertainty:

- the uncertainty in the true value of the parameters, and
- the uncertainty in the choice of the model (here, a rigid relation represented by a Gaussian).

5.1. Learning the Maximum Likelihood Model

Let us first consider the incremental learning of the Gaussian model parameters that maximize the likelihood of the relations observed in the video:

$$\hat{\mu}_t = \frac{\pi_{t-1}\hat{\mu}_{t-1} + w_t r_t}{\pi_{t-1}+w_t}, \tag{9}$$

$$\hat{\sigma}^2_t = \frac{\pi_{t-1}(\hat{\sigma}^2_{t-1} + (\hat{\mu}_t - \hat{\mu}_{t-1})^2) + w_t(r_t-\hat{\mu}_t)^2}{\pi_{t-1}+w_t}, \tag{10}$$

$$\pi_t = \pi_{t-1} + w_t, \tag{11}$$

where $\hat{\mu}_t$ and $\hat{\sigma}^2_t$ are the mean and the variance of the Gaussian, and π_t is the cumulative weight of preview observations (with $\pi_0 = 0$). To account for more or less reliable observations, the parameter updates are weighted by their likelihood product $w_t = p(z_{i,t}|x_{i,t})p(z_{j,t}|x_{j,t})$. For example, in the case of an occlusion or a loss of tracking, nothing

can be learned but the tracker will produce a (meaningless) observation nevertheless. We therefore discount such observations by using the observation likelihood as an indicator of reliability.

5.2. Uncertainty in the Parameters

When we have only a few observations, the Gaussian parameters produced by the maximum-likelihood estimation are uncertain. This uncertainty will decrease with the number of observations. We thus need to augment the variance of the informative part of the potential ψ (Eqn. 8) as a function of the number of observations. It turns out that the influence of the uncertainty in the mean is insignificant compared to the influence of the uncertainty in the variance; we therefore neglect the former. We consequently choose a variance $\tilde{\sigma}^2$ in a way that bounds the risk of underestimating the true variance, i.e., $P(\tilde{\sigma}^2 \leq \sigma^2) = \alpha$, where conventionally $\alpha = 0.95$. Since empirical estimates of variance follow a χ^2 distribution,

$$\tilde{\sigma}_t^2 = \frac{\pi_t}{\chi^2_{\pi_t - 1}(\alpha)} \hat{\sigma}_t^2, \tag{12}$$

where $\chi^2_{\pi_t - 1}(\alpha)$ is the inverse of the cumulative density function of the χ^2 distribution evaluated at probability α. Notice that the weight π_t is used instead of the number of observations, following the same reasoning as above: We do not want to excessively decrease the uncertainty in the model due to unreliable observations.

We can then simply define a new Gaussian model that takes the uncertainty in the parameters into account:

$$\psi_{i,j,t}(x_{i,t}, x_{j,t}) = \mathrm{e}^{-\frac{(r_t - \hat{\mu}_t)^2}{2\tilde{\sigma}_t^2}}. \tag{13}$$

5.3. Uncertainty in the Parametric Model

As motivated earlier, we are mainly interested in learning those relations that fit the chosen parametric model. To make best use of learned, informative relations during tracking while avoiding distractive bias due to observed relations not well represented by the model, we consider the latter to be completely uninformative and represent them by a uniform potential. In practice, nothing is black or white and a model may be more or less appropriate for the relation. Therefore, we represent the potential function by a weighted sum of the learned model and a uniform potential. The probability of observing a relation $r_{i,j,t} = x_{i,t} - x_{j,t}$ between features i and j at time t is then given by

$$\psi^+_{i,j,t}(x_{i,t}, x_{j,t}) = \lambda_t \mathrm{e}^{-\frac{(r_t - \hat{\mu}_t)^2}{2\tilde{\sigma}_t^2}} + (1 - \lambda_t)\frac{1}{2}, \tag{14}$$

where λ_t is the probability that the relation corresponds to a Gaussian model (see Fig. 1). To estimate λ_t, we introduce a

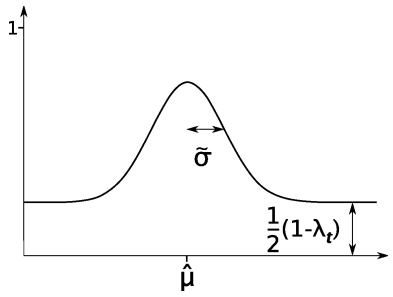


Figure 1. Example of an uncertain relation.

method inspired from the Kolmogorov-Smirnov test. Recall that the Kolmogorov-Smirnov distance is given by

$$K_n = \sqrt{n} \max_{-\infty < x < \infty} \left| \hat{F}(x) - F_n(x) \right|, \tag{15}$$

where n is the number of observations, $F_n(x)$ is the empirical cumulative distribution function of the n observations, and $\hat{F}(x)$ is the cumulative maximum-likelihood distribution. This distance is compared to a threshold, say, to classify a sample as Gaussian or non-Gaussian.

Our context is different: We are not interested in precise Gaussianity but in relations that are about as *predictive* as Gaussians. Therefore, the original Kolmogorov-Smirnov test is unsuitable in that its sensitivity grows without bounds with n. Instead of Eqn. 15, we use an expression independent of the number of observations,

$$D = \frac{1}{|I|} \int_I \left| \hat{F}(x) - F_n(x) \right| \mathrm{d}x, \tag{16}$$

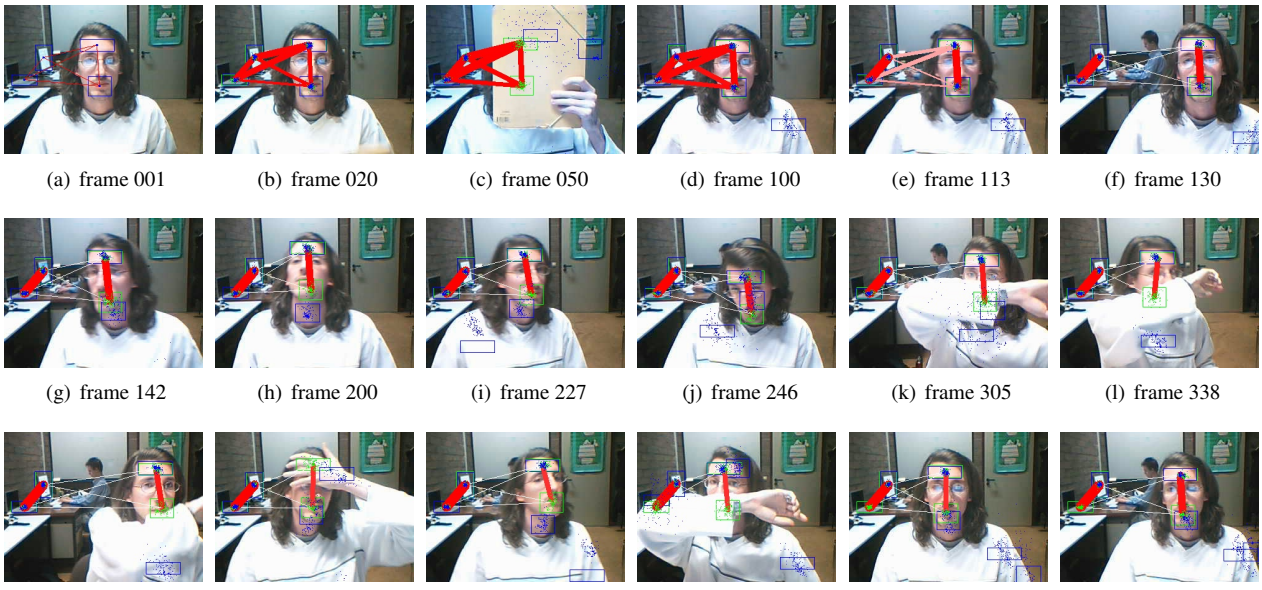
where I is the interval within which the two functions are compared. Notice that we use an integral instead of the maximum as it renders the measure both more robust and more discriminative in our context: Since it considers more than a single value it produces smoother curves, and outliers are more robustly detected because their cumulative effect is picked up by the integral.

To simplify matters, we assume that this distance D has a Gaussian distribution, which leads to the pseudo-probabilistic weighting function

$$\lambda_t = \mathrm{e}^{\frac{-D^2}{T_D^2}}, \tag{17}$$

where T_D is a user-settable parameter that represents the allowed deviation of observed relations from Gaussianity.

The resulting uncertain potential function $\psi^+_{i,j,t}(x_{i,t}, x_{j,t})$ (Eqn. 14) is plugged into Eqn. 7 for tracking. As we have seen, this results in a model of the relation that is not as informative as the maximum likelihood but exerts less counterproductive bias during tracking under non-Gaussian and non-stationary relations.

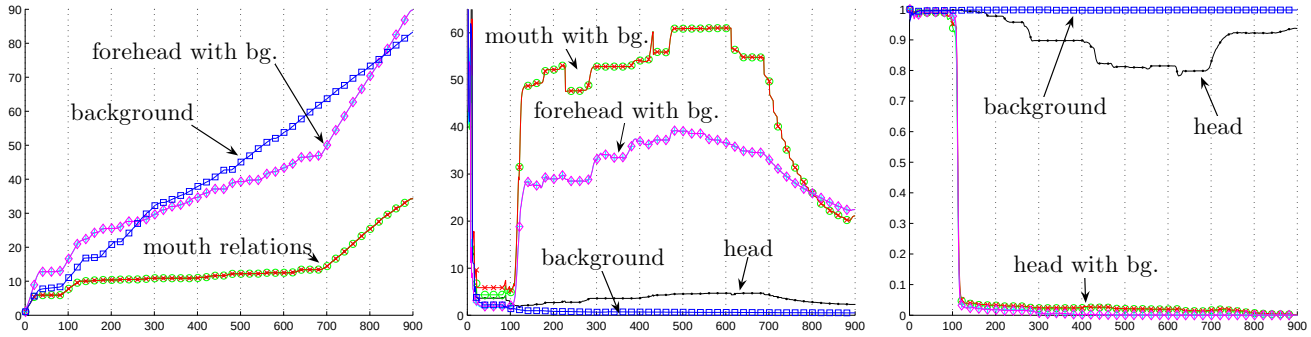


(a) frame 001 (b) frame 020 (c) frame 050 (d) frame 100 (e) frame 113 (f) frame 130

(g) frame 142 (h) frame 200 (i) frame 227 (j) frame 246 (k) frame 305 (l) frame 338

(m) frame 378 (n) frame 457 (o) frame 509 (p) frame 580 (q) frame 700 (r) frame 900

Figure 2. Representative results. Thick lines correspond to relations of low variance $\tilde{\sigma}$ and vice versa; red saturation is proportional to the probability λ_t. Each feature is tracked twice, with relations (green) and without (blue). Frame indices are given below each image.



(a) weight (b) variance (c) probability

Figure 3. Evolution of the relations. The two relations between mouth and background are superimposed, as are those between the forehead and the background. In (a), all three mouth relations are superimposed.

6. Experiments

In this section, we demonstate the performance of our method on a representative example of learning and tracking of 2D rigid relations. More results as well as videos can be found at *http://intelsig.montefiore.ulg.ac.be/˜declercq/*. To emphasize the contribution of the relations we chose to use a very simple feature descriptor. Features are represented by fixed image templates extracted from the first frame, and their likelihoods are computed using the sum of squared pixel differences. Their 2D coordinates are tracked with particle filters; no orientation or scale changes are considered. The informative part of the relations is represented by a Gaussian model for each 2D coordinate. We use $\sigma_\Delta = 5$ pixels and $T_D = 0.04$ pixels; these are the only user-settable parameters of our method.

Figure 2 shows the result sequence taken with a webcam at a resolution of 320×240 and processed on-line. Four features were selected by hand, two from the face and two from the background. Relations were also selected by hand. In this example, the relations are first learned as rigid because the scene is initially motionless. As seen in Figs. 2(b) and 3(a), the relations related to the mouth are learned more slowly than the others due to their lower likelihood

in the image. Once the head starts to move, the rigid relations connecting it with the background are rapidly unlearned. The probabilities λ_t of these relations become insignificant, and their variances increase. This clearly separates the graph into two subgraphs, one for the face and another for the background. Over the following frames, the face is successfully tracked despite the occlusions and its out-of-plane motions. Once the face–background relations have been detected as non-rigid, they do not influence the tracking anymore.

To illustrate the effect of the uncertain models on tracking, we track duplicate versions of the features without relations, represented in blue in Fig. 2. As the figure reveals, these features are tracked very poorly and have to be reinitialized many times during the sequence. It is thus clear that it would have been very difficult to learn a relational model from them without exploiting the – albeit uncertain – partially-learned relations from the start.

The end of the sequence (frames 700–900) is mostly motionless. Figure 3 shows that the probability of the forehead–mouth relation slowly increases and that all variances decrease. The probabilities of the relations between the facial features and the background do not increase because they were clearly non-rigid during the major part of the sequence. It will thus take much more time for their observation distributions to return to a Gaussian shape.

We implemented our algorithm on a Pentium Core 2 Duo 2×2 GHz. For a number of features between 4 and 10 with 3 relations each, it runs at between 8 and 20 frames per second.

7. Conclusions

In this paper we presented a new framework for on-line learning of feature graphs. This method is completely unsupervised and uses tracking to find correspondences between features. At the same time, information extracted from previous frames is immediately used to aid the tracking in the new frame. For representing this information, we proposed an uncertain model of relations based on a parametric model that incurs only a low computational cost. Several sources of uncertainty were identified and incorporated into the representation of the relations. The resulting uncertain model contributes stability to tracking without exerting overly strong, counterproductive bias.

The experiment demonstrates the ability of the uncertain model to assist tracking without biasing it, and – conversely – that tracking was essential for learning the uncertain model. The algorithm performed successfully under various difficulties such as occlusions, clutter and spurious connections between uncorrelated features.

In this paper, we presented the theory for the case of learning rigid, Gaussian relational models, but similar developments are possible for other parametric distributions. This work can be applied to the efficient learning of object models for various applications such as object detection, recognition, classification and tracking. We will explore some of these in future work.

8. Acknowledgment

This work is supported by a grant from the Belgian National Fund for Research in Industry and Agriculture (FRIA) to A. Declercq and by the EU Cognitive Systems project PACO-PLUS (IST-FP6-IP-027657).

References

[1] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *IJCV'04*, 56(3):221–255, 2004. 1, 2

[2] V. Comaniciu and P. Meer. Kernel-based object tracking. In *IEEE Trans. Pattern Anal. Machine Intell. vol 25*, pages 564–577, 2003. 1

[3] D. Crandall, P. F. Felzenszwalb, and D. P. Huttenlocher. Object recognition by combining appearance and geometry. In *Toward Category-Level Object Recognition*, pages 462–482, 2006. 1

[4] N. Dowson and R. Bowden. N-tier simultaneous modelling and tracking for arbitrary warps. In *BMVC'06*, volume 2, pages 569–578, 2006. 2

[5] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *IJCV'00*, 40(1):25–47, 2000. 2

[6] G. Hua and Y. Wu. Multi-scale visual tracking by sequential belief propagation. *CVPR'04*, 1:826–833, 2004. 3

[7] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *IJCV'98*, 29(1):5–28, 1998. 1

[8] M. Leordeanu and R. Collins. Unsupervised learning of object models from video sequences. *CVPR'05*, 1:1142–1149, 2005. 2

[9] J. Lim, D. A. Ross, R.-S. Lin, and M.-H. Yang. Incremental learning for visual tracking. In L. Saul, Y. Weiss, and L. Bottou, editors, *NIPS'05*, pages 801–808. 2005. 2

[10] T. Mathes and J. Piater. Robust non-rigid object tracking using point distribution models. In *BMVC'05*, pages 849–858, 2005. 2

[11] D. Ramanan, S. M.-D. A. Forsyth, and M.-K. Barnard. Building models of animals from video. *IEEE Trans. Pattern Anal. Machine Intell.*, 28(8):1319–1334, 2006. 2

[12] F. Scalzo and J. H. Piater. Unsupervised learning of dense hierarchical appearance representations. In *ICPR'06*, volume 2, pages 395–398, 2006. 1, 2

[13] L. Sigal, Y. Zhu, D. Comaniciu, and M. Black. Tracking complex objects using graphical object models. In *Proc. International Workshop on Complex Motion*, 2005. 1

[14] E. B. Sudderth, M. I. Mandel, W. T. Freeman, and A. S. Willsky. Visual hand tracking using nonparametric belief propagation. In *CVPRW'04*, volume 12, page 189, 2004. 1

[15] F. Tang and H. Tao. Object tracking with dynamic feature graph. In *VS-PETS'05*, pages 25–32, 2005. 1, 2