Geometric Layout Based Graphical Model for Multi-Part Object Tracking

Vijay Badrinarayanan, Francois Le Clerc, Lionel Oisel
Thomson R&D France
1 Avenue de Belle-Fontaine
35576 - Cesson Sevigne, France
{vijay.badrinarayanan}@thomson.net

Patrick Perez
INRIA Rennes - Bretagne Atlantique
Campus Universitaire de Beaulieu
35042 Rennes Cedex - FRANCE
perez@irisa.fr

Abstract

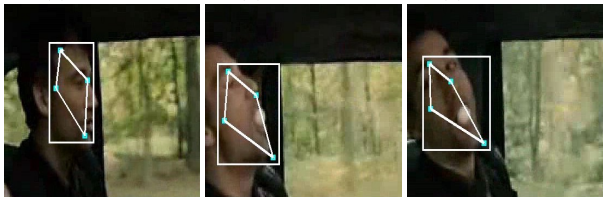
This work puts forth a probabilistic graphical framework to track unoccluded objects undergoing large out of image plane rotations and/or presenting large scale variations in video sequences. The proposed scheme incorporates measurements from an ensemble of local patch trackers and inter-patch geometric layout to arrive at a sample based approximation of the state posterior. Following this, the geometric layout is updated online using the Iterative Conditional Estimation technique. These steps are iterated until convergence to arrive at the final state posterior.
In contrast to offline training based schemes the proposed framework imposes no prior on the geometric layout and instead relies on online update of the geometric layout, thus broadening the scope of usage. Amongst other advantages, the scheme implicitly estimates the scale of the target and also adapts to varying target appearances to enable tracking under a fair degree of out of the image plane rotations. The tracking abilities of this scheme is put to test on several challenging videos with scale changes, out of the image plane rotations, illumination changes and motion jerks. Whereever possible qualitative comparisons are facilitated using videos from standard databases.

1. Introduction

The motivation for this contribution arises from the following two problems: In case an object (say face) undergoes large rotations in depth, then, the geometry interconnecting a set of patches on this object (say a polygon), termed geometric layout, is distorted. For visual tracking, this distortion is a cue to detect rotations in depth from monocular sequences. Thereby any changes in the geometric layout prior must be kept minimal. On the other hand when the target undergoes scaling, due to camera focal length changes or its own motion relative to the camera, this prior must adapt quickly to capture the right scale. Thus two problems need to be tackled;

1. How to differentiate between situations when the target object is undergoing 3D rotations or scaling.
2. How and how much should the prior be changed in each situation.

(a). Frame 1 (b). Frame 34 (c). Frame 62



(a). Frame 107 (b). Frame 172 (c). Frame 204

Figure 1. Illustration of results on the challenging *Children of Men* sequence. The little blue patches represent the centres of the local patch trackers, the rhombus represents the estimated geometric layout and the rectangle is a semi-precise bound.

Before proceeding to the main content of this paper, a brief review into the state of art is provided below.

Visual tracking algorithms vary in their degree of robustness against distractive measurements [8] and their ability to precisely outline the boundaries of the target object [10]. The state of art can be broadly classified into two categories possessing one of the above two qualities. Trackers working in *state spaces of small dimensions* like [2, 21] perform well under drastic illumination changes, short occlusions and small out of image plane (depth) rotations of the

target. Their appearance model updates range from complex holistic updates using foreground-background analysis [21], linear subspace or manifold based modeling [6, 12] to feature point resampling [2] with no regard to spatial geometry. Both of these categories lead to tracking drift over time. In addition, target scale is commonly approximated to one of a few discrete scales, as in [4, 6] or estimated using a parametric motion model which is often oversimplified. Thus techniques in this category are usually employed for *imprecise tracking* of a target over a lengthy duration of time. On the other hand *contour trackers* [10] accurately delineate the target boundary but need frequent user interaction to steer the unstable parts of the contour. Such trackers are usually employed over time periods of a few seconds. Another differentiating factor is that trackers in the first category rely on appearance models like color histograms/templates of the target to infer the *state* (location and scale) of the target at each new frame, whereas, the contour tracker is mainly driven by geometric cues and gradients at the occluding contours of the object.

The scheme presented in this paper relies upon local patch trackers [2] and their interconnecting geometry to output a *geometric layout* at each new frame from which the location, scale and semi-precise bounds (derived more formally than based on a *few* searches for the target position, scale in state space) of the target can be inferred directly. An illustration is provided in Fig. 1. The proposed scheme must not be misconstrued as a contender for shape/contour tracking. It is based on rough geometric layouts as opposed to precise contour based tracking as in [23]. As the scheme is designed to work with unconstrained sequences, it does not rely on offline background modeling as used by Sullivan et al [19].

The use of spatial geometry information has been exploited earlier in conjunction with learning, in contributions such as [18] where the problem is to locate several features on a face when some are occluded. A tracking example using rigid geometric cues in combination with color is found in [15]. Other examples in recent literature using distributed set of trackers include the work by Yang et al. [22], where auxiliary objects are tracked to infer the location of the principal target. Harini et al. [20] use a set of templates linked in a geometric fashion to track moving vehicles of very small dimension. Pilet et al [16] aim at deformable object detection using mesh models and feature point correspondences. They, however, deal with a detection problem and tests are made on laboratory sequences. Problems like face alignment only consider minor rotations in depth of mug-shot faces and approaches like the one proposed by Liu [13] requires extensive training of weak classifiers for computing mesh alignment. Tracking in unconstrained sequences cannot rely largely on offline learning. Finally, none of the preceeding techniques attempt to deal specifically with the difficult issue of rotations in depth in unconstrained sequences. A second issue is target scale estimation. The most common technique is approximation to one of a few discrete scales as in [5, 6]. Other techniques include imposing a simple affine motion constraint on the object and estimating the model parameters by tracking/matching a few feature points. To achieve meaningful results the features must be *well distributed* over the target and even so, when there is combined depth rotation and scaling such a model is oversimplified. Beyond this it is also necessary to adapt the appearance model of the target during scaling.

The aim of this work is to develop a tracking scheme that *implicitly* accomodates changing object appearances due to rotations in depth (measuring rotations in depth in arbitrary sequences with no known camera parameters is a difficult problem) and follows scale changes of the object without explicit motion model estimation (the accuracy of model estimates is controlled by unknown measurement noise). Motivated by these aims a concerted participation of an *ensemble of local patch trackers* using their individual measurements/observations and geometric layout is investigated. A probabilistic graphical framework is formulated to analyse these *distributed measurements* and the important problem of *geometric layout update*.

The following Section 2 describes the probabilistic graphical modeling of the problem. Section 3 delineates the characteristics of the local observations using patch trackers and elaborates the proposed algorithm. Section 4 details the experimental setup. The results of the experiments, qualitative comparisons and discussions regarding the strengths and drawbacks of the approach form Section 5. Pointers to extension and prospective work are given in Section 6. Conclusions are drawn in Section 7.

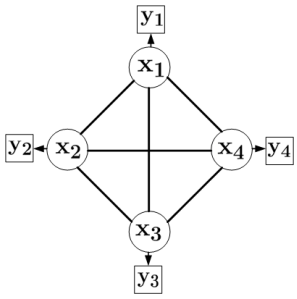


Figure 2. Graphical model. $\{x_i,\ i = 1 : 4\}$ are the hidden state variables and $\{y_i,\ i = 1 : 4\}$ are their corresponding observations.

2 Problem Description

Consider the fully connected graphical model shown in Fig. 2. The joint hidden state variable is denoted as

$X = \{x_i,\, i\epsilon\mathcal{V}\}$ and the corresponding set of observations denoted as $Y = \{y_i,\, i\epsilon\mathcal{V}\}$, where $\mathcal{V}$ is the set of nodes (vertices) in the graph. The joint posterior of the hidden state and the observation, parameterized by $\theta, \mathcal{P}_\mathcal{R}$, is taken to have the following form.

$$p\big(X|Y;\theta,\mathcal{P}_\mathcal{R}\big) \propto l_t\big(Y|X\big) l_\mathcal{P}\big(X|\mathcal{P}_\mathcal{R}\big) \prod_{(i,j)\epsilon\Gamma} \Psi\big(x_i, x_j|\theta\big), \tag{1}$$

where $l_t\big(Y|X\big)$ is a joint likelihood function; the prior is factored into a geometric layout similarity function $l_\mathcal{P}\big(X|\mathcal{P}_\mathcal{R}\big)$ with *fixed parameter* $\mathcal{P}_\mathcal{R}$ and pairwise potential (or compatibility) functions $\big\{\Psi\big(x_i, x_j|\theta\big),\, (i,j)\,\epsilon\Gamma\big\}$ with Γ denoting the set of all edges in the graph. It is to be noted that the set of pairwise potential functions represent the geometric layout and from here on the term geometric potentials will be used interchangeably with geometric layout.
Given the data $Y = y$ at some instant in the video sequence, the objective is to infer the marginal posteriors $\big\{p\big(x_i|Y = y;\theta,\mathcal{P}_\mathcal{R}\big),\, i\epsilon\mathcal{V}\big\}$ or *beliefs* at each node in the set $\mathcal{V}$, of the graph. Some interesting algorithms including Belief/Loopy-Belief Propagation have been introduced in recent literature by [18, 7] for such inferences. The assumption underlying these algorithms is that a *prior* (potential function) on the joint state variable has been learnt offline via extensive training. In an unconstrained tracking context such priors are difficult to obtain and even so, they are not suitable when there is large *scaling* and *rotations* in depth of the object. Therefore these priors must be updated *online* in conjunction with the likelihood.
A direct application of the iterative Expectation Maximisation (EM) algorithm is not suitable as the normalization factor or partition function of the joint posterior in Eqn. 1 is difficult to obtain in closed form in general cases. A Monte-Carlo sampling based alternative is then sought after to approximate the posterior given an estimate of θ. The parameter θ is then updated using the Iterative Conditional Estimation (ICE) technique proposed by [17]. This *non-sequential* iterative algorithm is detailed in the following section.

3 Proposed Algorithm

A set of local patches are tracked from one frame to the next using some convenient patch tracker (the details of patch tracking are presented in Sec. 4). The measurement or observation model associated to this patch tracking is as given below.

$$y_i = \underset{x_i}{\operatorname{argmin}}\, \mathcal{D}\big[f\big(x_i\big), f^*\big] + \eta, \tag{2}$$

where $\mathcal{D}$ is a distance function, typically template/patch cross-correlation or distances between color histograms. $f\big(x_i\big)$ is the extracted local patch and f^* is a reference model for the tracked patch and η is the corrupting noise whose statistics are unknown. The non-linearity of the above model and the unknown characteristics of the noise makes it impossible to define an analytical form for the measurement density conditional on the state. The standard alternative then is to define a measurement likelihood function. In the proposed scheme the following Gaussian form of measurement likelihood is defined and utilised.

$$l_t\big(Y|X\big) = \frac{1}{Z} \prod_{i\epsilon\mathcal{V}} \mathcal{N}\big(x_i; \underset{x_i}{\operatorname{argmin}}\, \mathcal{D}\big[f\big(x_i\big), f^*\big], \Sigma_i\big), \tag{3}$$

where the subscript t emphasizes the fact that it is derived from patch trackers, Z, the normalization is known and Σ_i is an empirically determined variance. The pairwise potential functions describing the relationship between the local state variables are assumed Gaussian as shown below;

$$\Psi\big(x_i, x_j\big) = \mathcal{N}\big(|x_i - x_j|; \mu_{ij}, \Sigma_{ij}\big),\, (i,j)\,\epsilon\Gamma, \tag{4}$$

by which the parameter $\theta = \{\mu_{ij}, \Sigma_{ij};\, (i,j)\,\epsilon\Gamma\}$. A note of interest here: the *form of the potential is not a multi-variate Gaussian*, as it is based on the absolute difference of state variables. This clearly brings out the inadequacy of attempting to learn this type of prior offline; as even a simple scaling of the target would affect the learnt potential considerably. Interestingly most algorithms in the tracking literature based on graphical models [18, 3, 11] do not attempt to deal with situations were the potentials undergoes scaling.

3.1 Importance Sampling Approximation of the Joint Posterior

Given a new frame in the sequence and an initial estimate of the parameter θ_{k-1}, at iteration $k-1$, the joint posterior in Eqn. 1 is approximated using samples drawn from a suitable importance sampling density constructed using l_t as described below.

Developing a proposal density:
Letting aside the geometric layout based prior which makes sampling unpractical, consider the following model of the joint posterior parameterized by the estimate θ_{k-1};

$$g\big(X|Y;\theta_{k-1}\big) = l_t\big(Y|X\big) \prod_{(i,j)\epsilon\Gamma} \Psi\big(x_i, x_j|\theta\big). \tag{5}$$

Probabilistic inference on this model using standard Belief propagation results in marginal posteriors $\big\{g\big(x_i|Y;\theta_{k-1}\big),\, i\epsilon\mathcal{V}\big\}$ or beliefs, denoted as $\big\{\mathbf{b}_{k-1}\big(x_i\big), i\epsilon\mathcal{V}\big\}$. It is to be emphasized that Belief propagation is necessary since the priors are not multivariate Gaussian. A proposal density is then developed

using these beliefs as;

$$q\big(X|Y;\theta_{k-1}\big) = \prod_{i\epsilon\mathcal{V}} \mathbf{b}_{k-1}\big(x_i\big). \tag{6}$$

Samples $\big\{X^s \sim q\big(X|Y;\theta_{k-1}\big),\ s = 1:M\big\}$ are drawn and the unnormalized importance sampling weights w^s computed as shown below:

$$w^s = \frac{l_t\big(Y|X^s\big) l_{\mathcal{P}}\big(X^s|\mathcal{P}_{\mathcal{R}}\big) \prod_{(i,j)\epsilon\Gamma} \Psi\big(x_i^s, x_j^s|\theta\big)}{\prod_{i\epsilon\mathcal{V}} \mathbf{b}\,(x_i^s)}. \tag{7}$$

Following the above computation, the joint posterior parameterized by estimate θ_{k-1} can be approximated as,

$$p\big(X|Y;\theta_{k-1},\mathcal{P}_{\mathcal{R}}\big) \approx \sum_{s=1:M} \widetilde{w}^s \delta_{X^s}\big(X\big), \tag{8}$$

where $\widetilde{w}^s = \frac{w^s}{\sum_{s=1:M} w^s}$ is the normalized importance weight.

In Eqn. 8 it is assumed that the geometric layout similarity function in the prior, $l_{\mathcal{P}}\big(X|\mathcal{P}_{\mathcal{R}}\big)$, can be evaluated pointwise. Apart from this fact this function merits special attention.

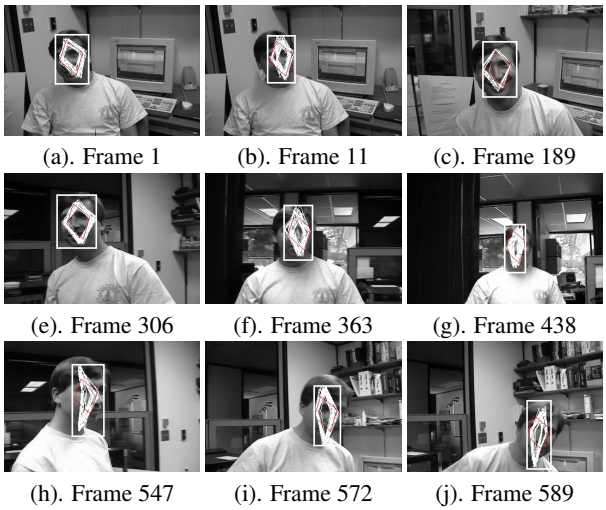
The Geometric Layout Similarity Function:
In this paper geometric layouts are in the form of *polygons*. At the initialization frame for the tracking sequence the polygon connecting the patch trackers is stored as a *reference polygon* $\mathcal{P}_{\mathcal{R}}$ for the target. A polygon represented by sample X^s, denoted as $\mathcal{P}\left(X^s\right)$ is compared with this reference to derive a measure of similarity for this sample. This matching is done using the standard polygon matching algorithm prescribed by Arkin et al [1], using polygon turning angle based polygon coding; which outputs a $\mathcal{L}_2$ distance between the *representational codes* of $\mathcal{P}_{\mathcal{R}}$ and $\mathcal{P}\left(X^s\right)$. The matching is in practice invariant to rotation to a reasonable extent. The geometric layout similarity *function* is then defined as shown below:

$$l_{\mathcal{P}}\big(X^s|\mathcal{P}_{\mathcal{R}}\big) \triangleq \frac{1.0}{\|\mathcal{P}_{\mathcal{R}} - \mathcal{P}\left(X^s\right)\|}. \tag{9}$$

The sample based joint posterior approximation can be visualized in Fig. 3. The joint state samples shown are true samples drawn from the sample based approximate. Therefore their spatial spread provides an insight into the uncertainty of the posterior at that frame.
The following step is to update the potential functions given the new data.

3.2 Parameter Update

The Iterative Conditional Estimation (ICE) technique prescribes parameter estimation given known state X and



(a). Frame 1 (b). Frame 11 (c). Frame 189
(e). Frame 306 (f). Frame 363 (g). Frame 438
(h). Frame 547 (i). Frame 572 (j). Frame 589

Figure 3. Tracking the *Jepson and Fleet* sequence. 30 joint state samples from the approximated joint posterior are displayed to give an insight into the uncertainty of the posterior. The red polygon is the unscaled reference model superimposed on each frame.

an instance of the observation $Y = y$. The essence of this technique lies in the following iteration;

$$\theta_k = \mathbf{E}_{p\left(X|Y;\theta_{k-1},\mathcal{P}_{\mathcal{R}}\right)} \Theta\big(X, Y\big), \tag{10}$$

where $\Theta\big(X, Y\big)$ is a statistical estimator of θ given X and Y. Substituting the MonteCarlo (MC) approximation from Eqn. 8 in Eqn.10 leads to a sample based approximation as shown below:

$$\theta_k \approx \sum_{s=1:M} \widetilde{w}^s \Theta\big(X^s, Y\big). \tag{11}$$

In the experiments reported in this paper, $\Theta\big(X, Y\big)$ is taken to be the Maximum Likelihood Estimator (MLE);

$$\tilde{\theta} = \underset{\theta}{\operatorname{argmax}}\, p\big(X|Y;\theta,\mathcal{P}_{\mathcal{R}}\big). \tag{12}$$

In the Gaussian case experimented with here, Eqn.10 leads to parameter estimation of the form,

$$\mu_{ij} = \sum_{s=1:M} \widetilde{w}^s |x_i^s - x_j^s| \ \ (i,j)\,\epsilon\Gamma. \tag{13a}$$

$$\Sigma_{ij} = \sum_{s=1:M} \widetilde{w}^s \big(|x_i^s - x_j^s| - \mu_{ij}\big)^T \big(|x_i^s - x_j^s| - \mu_{ij}\big), \ \ (i,j)\,\epsilon\Gamma. \tag{13b}$$

The steps described in Sections 3.1 and 3.2 can be iterated until convergence. Fig. 4 provides the implementation of this algorithm used in the experiments reported in this paper. The next section focuses on the experiments to test this algorithm.

Algorithm: At frame n;

1. **Initialization:**
   $\theta_0 = \left\{\mu_{ij}^{n-1}, \Sigma_{ij} = \mathcal{Q};\ (i,j)\,\epsilon\Gamma\right\}.$
   $l_t(Y|X) = \frac{1}{Z}\prod_{i\epsilon\mathcal{V}} \mathcal{N}\left(x_i; \text{argmin}_{x_i}\, \mathcal{D}\left[f(x_i), f^*\right], \Sigma_i\right).$

   where, the diagonal covariances $\mathcal{Q}, \{\Sigma_i,\ i\epsilon\mathcal{V}\}$ are empirically set (See Section 4).
   $k \longleftarrow k-1:$

2. **Importance Sampling Approximation of Posterior**

   Beliefs derived from Belief Propagation

   $$\left[\widetilde{\Sigma}_i\right]^{-1} = \left[\Sigma_i\right]^{-1} + \textstyle\sum_{j\epsilon\mathcal{V}-i}\left[\Sigma_j + \mathcal{Q}\right]^{-1},\ i\epsilon\mathcal{V}$$
   $$\widetilde{\mu}_i = \widetilde{\Sigma}_i\left[\left[\Sigma_i\right]^{-1}\mu_i + \textstyle\sum_{j\epsilon\mathcal{V}-i}\left[\Sigma_j+\mathcal{Q}\right]^{-1}\left[\mu_j+\mu_{ji}\right]\right],\ i\epsilon\mathcal{V}$$
   $$\mathbf{b}_{k-1}(x_i) = \mathcal{N}\left(x_i; \widetilde{\mu}_i, \widetilde{\Sigma}_i\right),\ i\epsilon\mathcal{V}.$$

   Proposal Density and Samples

   $$q\left(X|Y;\theta_{k-1}\right) = \textstyle\prod_{i\epsilon\mathcal{V}} \mathbf{b}_{k-1}(x_i).$$
   $$\left\{X^s \sim q\left(\frac{X}{Y,\theta_{k-1}}\right),\ s = 1:M\right\}.$$

   Importance Weights and posterior

   $$w^s = \frac{l_t\left(Y|X^s\right) l_{\mathcal{P}}\left(X^s|\mathcal{P}_{\mathcal{R}}\right) \prod_{(i,j)\epsilon\Gamma} \Psi\left(x_i^s, x_j^s|\theta\right)}{\prod_{i\epsilon\mathcal{V}} \mathbf{b}\left(x_i^s\right)}.$$
   $$p\left(X|Y;\theta_{k-1},\mathcal{P}_{\mathcal{R}}\right) \approx \textstyle\sum_{s=1:M} \widetilde{w}^s \delta_{X^s}(X),$$
   where $\widetilde{w}^s = \frac{w^s}{\sum_{s=1:M} w^s}$.

3. **Parameter Update**:
   $\mu_{ij} = \sum_{s=1:M} \widetilde{w}^s |x_i - x_j|,\ (i,j)\,\epsilon\Gamma$
   $\Sigma_{ij} = \sum_{s=1:M} \widetilde{w}^s \left(|x_i - x_j| - \mu_{ij}\right)^T \times$
   $\left(|x_i - x_j| - \mu_{ij}\right),\ (i,j)\,\epsilon\Gamma.$

Set $\mu_{ij}^n = \mu_{ij}$.

Figure 4. Implemented verison of the proposed algorithm

4 Experiments

The video sequences used in all the experiments had targets undergoing significant rotations in depth and/or large scale changes. In addition poor recording quality, rapid changes in illumination and motion jerks are also found in some sequences (See 8). Occlusion handling is currently beyond the scope of this paper and therefore the *targets are unoccluded* in all the sequences.

Quantitative comparisons are difficult to perform for the proposed approach as their are no direct contenders specifically dealing with tackling the issue of rotations in depth in unconstrained sequences. Further, due to non-availability of executables for related state of art, to the extent possible, standard sequences are used to aid visual judgement. Standard video sequences like the Fleet sequence [9], the Behzad sequence and the deflating Balloon sequence were used for testing the scheme. The sources for these sequences and *comparable results* for a few of them, which include techniques like adaptive template matching, are pointed to below the appropriate results. Further challenging video sequences like in Figs. 1, 8 have also been experimented with to test the efficacy of the approach. Relevant *supplementary material* in the form of videos are also provided for review.

Nature of a Patch and Patch Tracking:

A patch is constructed by a set of, *possibly overlapping*, feature point templates. It is emphasized that a patch can be of any *arbitrary shape* and is seldom rectangular. An example patch constructed using 4 feature point templates is shown in Fig. 5. Patch tracking involves tracking each feature point in its set and using these tracking results to arrive at the estimated location of the whole patch. A subset of these points (thereby their templates) are replaced online as the need arises and therefore the patch is said to *evolve* over time. An illustrative procedural detail of patch tracking is provided in Fig. 5 alongwith additional explanations. The initial variance of the potentials is empirically fixed as,

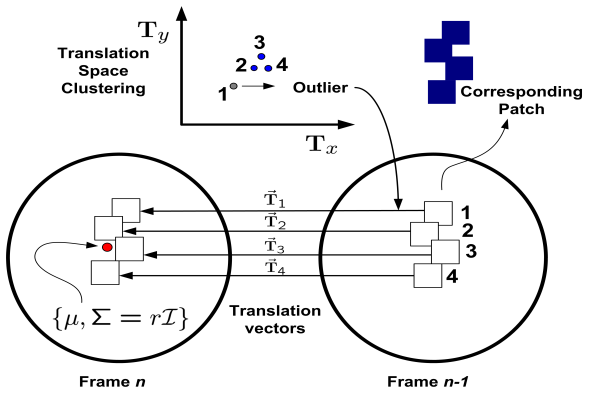


Figure 5. Patches and Patch Tracking. A patch, see top right hand corner, is constructed using $F = 4$ feature points each with templates of size 21×21. Templates are marked as small rectangles. Normalized Cross-Correlation is used for point tracking and translation space clustering for outlier detection. The empirical likelihood parameters are indicated inside the diagram, with $r = 13.0$.

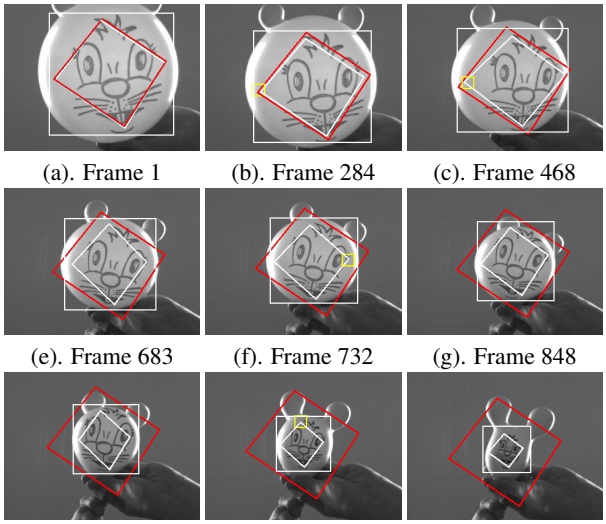
$\mathcal{Q} = \lambda\mathcal{I}_{2\times 2}$ (See Fig. 4) with $\lambda = 26.0$ for numerical compatibility with the likelihood variances (See Fig. 5). At each frame the algorithm was run for 1 iteration as the qualitative result was seen to be sufficient.

Patch Selection:

Patches are manually selected at the periphery of the tracked

object in a way that the layout of the patches are isomorphic to the general shape of the object, for instance, an elliptical layout for human faces. The number of patches are varied depending on the shape of the layout and the computational power at ones disposal.

5 Results and Discussions

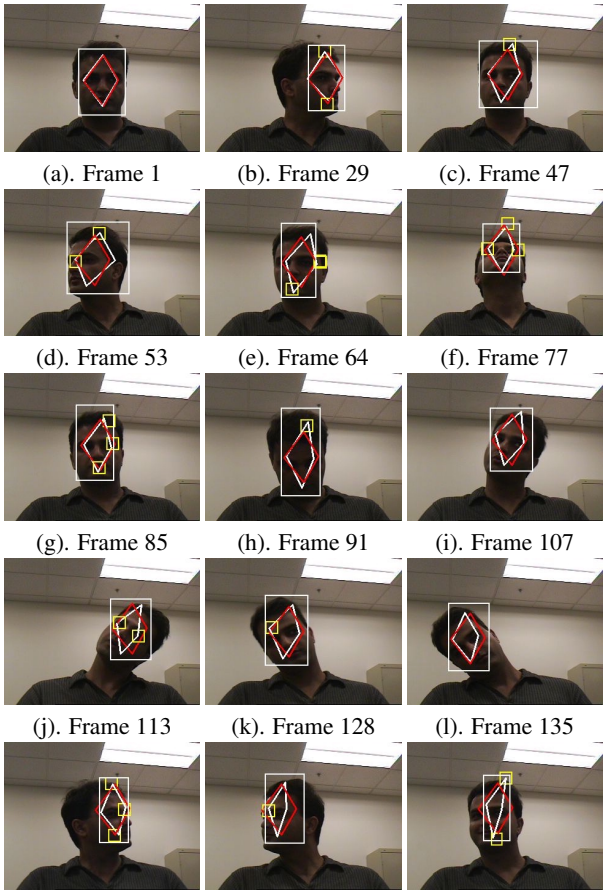


(a). Frame 1 (b). Frame 284 (c). Frame 468

(e). Frame 683 (f). Frame 732 (g). Frame 848

(h). Frame 956 (i). Frame 1000 (j). Frame 1080

Figure 6. Results on the *Deflating Balloon*(Mouse) sequence. The white polygon represents the estimate of the geometric layout derived as the mean of the joint posterior. Newly sampled feature points are displayed in small yellow rectangles. Source video can be found at http://esm.gforge.inria.fr/ESMdownloads.html



(a). Frame 1 (b). Frame 29 (c). Frame 47

(d). Frame 53 (e). Frame 64 (f). Frame 77

(g). Frame 85 (h). Frame 91 (i). Frame 107

(j). Frame 113 (k). Frame 128 (l). Frame 135

(m). Frame 154 (n). Frame 169 (o). Frame 178

Figure 7. Results on the Honda/UCSD Database *Behzad1* test sequence. Source video can be found at http://vision.ucsd.edu/leekc/HondaUCSDVideoDatabase/HondaUCSD.html.

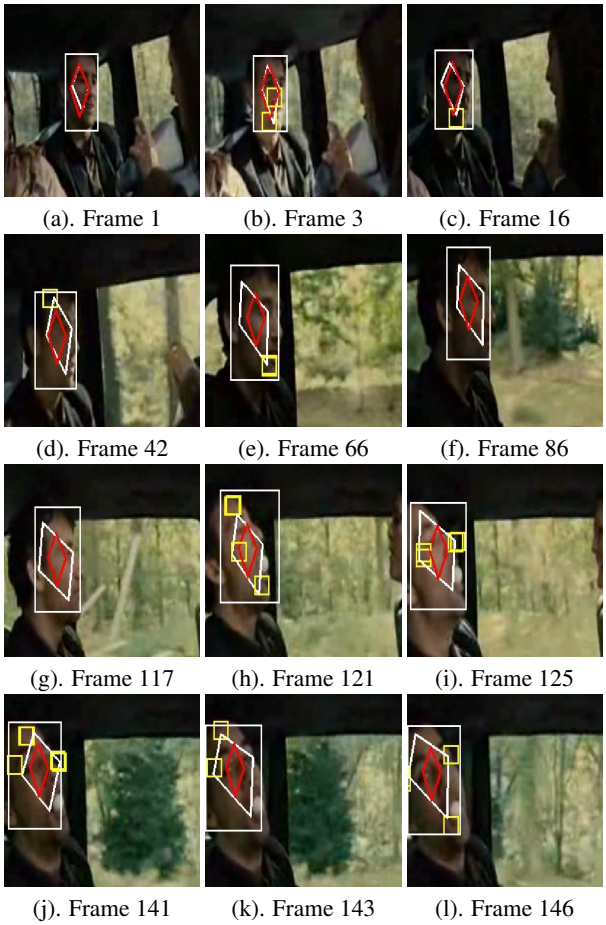
The results presented in this section are roughly arranged in an *increasing order of difficulty*, starting from contrived lab videos to outdoor cinema sequences. The first sequence shown in Fig.6 is a deflating and deforming balloon sequence presenting large scale change but with little rotation in depth. See Malis [14] for comparable results.

Feature Point Rejection and Resampling:
In several frames in Fig. 6 feature points at some patch need to be eliminated and replaced in order to adapt to deformations or rotations in depth. This is an important issue which controls the efficacy of the proposed scheme to adapt to changing appearances. In this scheme the outlier feature points (See Fig. 5 for an explanation of how outliers are marked) in each patch are eliminated from the F points and a new subset of replacement feature points, denoted $\{F_i^R,\ i\epsilon\mathcal{V}\}$ are sampled/resampled from corresponding densities:

$$F_i^R \sim \mathcal{N}\big(x_i; \mathbf{E}_{p\left(x_i|Y;\theta,\mathcal{P}_\mathcal{R}\right)}\delta_{x_i^s}(x_i), \mathcal{S}\big),\ i\epsilon\mathcal{V} \qquad (14)$$

where the expectation is evaluated using the MC approximation of the belief and $\mathcal{S}$ is a fixed sampling covariance (diagonal) matrix (It is set to $3.0\mathcal{I}_{2\times 2}$ in the experiments).

Patch Evolution:
An important point arises in this context of feature point resampling. When new feature points are sampled, new templates centered on these points are associated to them; as a consequence of which the patch composition changes. These *new templates* correspond to *new parts* of the target. Therefore, in a patch, some templates could possibly be from distant past (if a particular point in the patch has been consistently tracked for a long duration), some relatively new and others new. Clearly, as this set of templates also form the reference model for the patch, resampling feature points (thereby their templates) means changing this reference model. This change is is essential to capture appearing parts of the target to offset tracker drift to an extent.



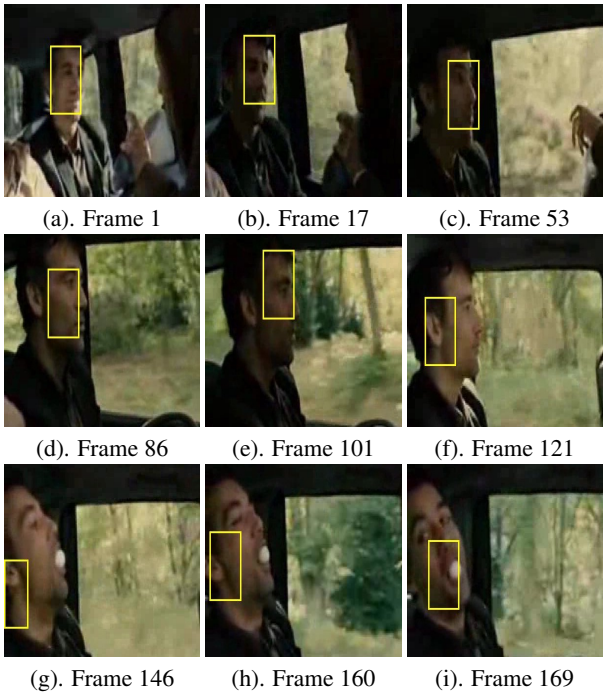
(a). Frame 1 (b). Frame 3 (c). Frame 16

(d). Frame 42 (e). Frame 66 (f). Frame 86

(g). Frame 117 (h). Frame 121 (i). Frame 125

(j). Frame 141 (k). Frame 143 (l). Frame 146

Figure 8. Results on the *Children of Men* sequence. This video has poor recording quality. Large motion jerks eventually cause tracking failure.



(a). Frame 1 (b). Frame 17 (c). Frame 53

(d). Frame 86 (e). Frame 101 (f). Frame 121

(g). Frame 146 (h). Frame 160 (i). Frame 169

Figure 9. Qualitative comparison with a color based particle filter of [15]. The frequent sliding of the estimated position is clearly noticeable.

The Behzad1 test sequence from the Honda/UCSD database in Fig. 7 presents a greater challenge to the proposed algorithm with frequent in-plane and out of plane rotations, scale changes and fast motions. Frequent resampling of feature points can be observed in several frames, which supports intuition. The proposed tracker is able to follow the key part of the face for the most part of the sequence. The scale of the target is also assessed reasonably well (See Figs. 7 (f) and 7(h)). In Figs. 7(f) and 7(m) new features are sampled at the threshold between the target and the background, but the tracker does not drift due to these feature points thanks to the geometric layout context built into the system. In spite of the foregoing advantages the tracking quality detoriates after several rotations, primarily due to a poor proposal density warranted by a non-sequential (*static*) approach and dependence only on the likelihood at that frame. This leads to unwarranted new feature points causing imperfections (See Fig.7(p)). The algorithm also lacks an estimate of the orientation of the subject.

The ability of the tracker to perform on poor quality videos is tested in Fig. 8. The tracker stays focused on the target despite large illuminations changes caused by shadowing permitted by normalized cross-correlation based feature point tracking and feature resampling. A 200 particle color based filter of [15] with no color model adaptation is distracted by this frequent change in illumination as seen in Fig. 9. The problem of *holistic* color model adaptation is indeed a difficult one; contrast that with the proposed scheme where the feature point resampling aids *partial* adaptation implicitly. The tracker also performs well in presence of some out of plane rotations due to target pose changes and ego-motion of the camera. The implicit following of scale changes is brought out in this sequence. In comparison scale estimation is generally *decoupled* from position estimation in particle filters due to dimensionality problems. Between 30-50 joint state samples were used in the experiments and the computational time for the tracker on a 2GHz CPU machine was estimated to be in the order of 10-12 frames/sec.

6 Prospective Work

The proposed algorithm has several future prospects. It is free from any topological constraints on the graphical model that can be imposed; therefore other graphs modeling a different conditional independency structure may be experimented with. The Gaussian nature of the geometric potentials can also be relaxed to include non-Gaussian potentials. The algorithm can also be extended to sequential estimation which can include predictive priors for the geometric layout. The experimentalist is also at liberty to choose different likelihoods to develop a better proposal density, in particular, including the geometric layout similarity should enhance performance. Although most results shown in this paper are on human faces, the algorithm inherently is free from any prior on the kind of object it can track.

The algorithm's use in arbitrary environments is still restricted due to the absence of occlusion handling capabilities. An immediate extension which can be envisaged is the introduction of color based holistic observations to recover from occlusions.

7 Conclusion

The key formalism presented in this paper is a probabilistic graphical framework for fusion of geometric layout contexts and local patch tracking. The fusion is performed in a two step manner. Firstly the results of patch tracking, configured as likelihoods, aid the approximation of the joint posterior of all the state variables, which is otherwise a difficult problem. In the second step this approximate is used to estimate online the potential functions describing the inter-patch geometric relationships. It is demonstrated with various examples that such a scheme is efficient in tracking under large out of plane rotations and scale changes of the target. In this setup, the difficult problem of target appearance model adaptation is dealt using local patch evolutions bearing upon the approximation of the joint posterior. The prior free nature of the algorithm makes it ideally suited for scenarios where priors on the graphical models cannot be easily learnt via training, as in unconstrained tracking problems, and therefore need to be updated online. A sequential extension to this approach holds future prospects.

References

[1] E. Arkin, L. Chew, D. Huttenlocher, K. Kedem, and J. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Trans. on PAMI*, 13(3):209–216, 1991. 4

[2] V. Badrinarayanan, P. Perez, F. L. Clerc, and L. Oisel. Probabilistic color and adaptive multi-feature tracking with dynamically switched priority between cues. In *ICCV, Rio de Janeiro, Brazil*, 2007. 1, 2

[3] M. Briers, A. Doucet, and S. S. Singh. Sequential auxiliary particle belief propagation. In *Proc. Fusion*, 2005. 3

[4] D. Comaniciu and P. Meer. Mean shift analysis and applications. In *ICCV, Kerkyra, Greece*, 1999. 2

[5] B. Han, Y. Zhu, D. Comaniciu, and L. Davis. Kernel-based bayesian filtering for object tracking. In *CVPR, San Diego, CA*, 2005. 2

[6] J. Ho, K.-C. Lee, M.-H. Yang, and D. Kriegman. Visual tracking using learned subspaces,. In *CVPR, Washington, DC*, 2004. 2

[7] G. Hua and Y. Wu. Multi-scale visual tracking by sequential belief propagation. In *CVPR, Washington, DC*, 2004. 3

[8] G. Hua, Y. Wu, and Z. Fan. Measurement integration under inconsistency for robust tracking. In *CVPR, New York City, NY*, 2006. 1

[9] A. Jepson, D. Fleet, and T. El-Maraghi. Robust online appearance models for visual tracking. *IEEE Trans. on PAMI*, 25(10):1296–1311, 2003. 5

[10] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *IJCV*, 1(4):321–331, 1987. 1, 2

[11] Z. Khan, T. Balch, and F. Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *ECCV, Copenhagen*, 2004. 3

[12] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Visual tracking and recognition using probabilistic appearance manifolds. *CVIU*, 99:303–331, 2005. 2

[13] X. Liu. Generic face alignment using boosted appearance model. In *CVPR, Minneapolis, Minnesota*, 2007. 2

[14] E. Malis. An efficient unified approach to direct visual tracking of rigid and deformable surfaces. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, USA*, 2007. 6

[15] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *ECCV, Copenhagen, Denmark*, 2002. 2, 7

[16] J. Pilet, V. Lepetit, and P. Fua. Real-time non-rigid surface detection. In *CVPR, San Diego, CA*, 2005. 2

[17] F. Salzenstein and W. Pieczynski. Unsupervised bayesian segmentation in hidden markovian fields. In *ICASPP, Detroit*, 1995. 3

[18] E. B. Sudderth, A. T. Ihler, W. T. Freeman, and A. S. Willsky. Nonparametric belief propagation. In *CVPR, Madison, Wisconsin*, 2003. 2, 3

[19] J. Sullivan, A. Blake, M. Isard, and J. MacCormick. Object localization by bayesian correlation. In *ICCV, Kerkyra, Corfu, Greece*, 1999. 2

[20] H. Veeraraghavan, P. Schrater, and N. Papanikolopoulos. Robust target detection and tracking through integration of motion, color and geometry. *CVIU*, 103:121–138, 2006. 2

[21] M. Yang and Y. Wu. Tracking non-stationary appearances and dynamic feature selection. In *CVPR, San Diego, CA*, 2005. 1, 2

[22] M. Yang, Y. Wu, and S. Lao. Intelligent collaborative tracking by mining auxiliary objects. In *CVPR, NewYork*, 2006. 2

[23] X. S. Zhou, D. Comaniciu, and A. Gupta. An information fusion framework for robust shape tracking. *IEEE Trans. on PAMI*, 27(1):115–129, 2005. 2