A probabilistic graph-based framework for plug-and-play multi-cue visual tracking

Shimrit Feldman-Haber [*], Yosi Keller[†]

Abstract

In this work we propose a novel approach for integrating multiple tracking cues within a unified probabilistic graph-based Markov Random Fields (MRF) representation. We show how to integrate temporal and spatial cues relating to unary and pairwise probabilistic potentials. As the inference of such high order MRF models is known to be NP-hard, we propose an efficient spectral relaxation based inference scheme. The proposed scheme is exemplified by applying it to a mixture of five tracking cues, and is shown to be applicable to wider set of cues. This paves the way for a modular plug-and-play tracking framework that can be easily adapted to diverse tracking scenarios. The proposed scheme is experimentally shown to compare favorably with contemporary state-of-the-art schemes, and provides accurate tracking results.

1 Introduction

Object tracking in video sequences is a challenging problem in computer vision, that is of significant applicative importance in this day and age, where metropolises such as London and New York are overlaid with large scale video surveillance systems. A gamut of approaches have been proposed over the years that aim to track an object $\mathcal{O}$ moving over the background $\mathcal{B}$ in a video sequence $\{\mathbf{F}_t\}$.

In this work we propose a unified probabilistic approach for integrating multiple tracking cues in a statistical inference scheme based on spectral graph matching. For that we propose to represent the video frames via sets of superpixels (SPs) [23, 27, 20] (Section 2.1), as this allows a robust statistical

[*]Faculty of Engineering, Bar Ilan University, Israel, shimritf@gmail.com.

[†]Faculty of Engineering, Bar Ilan University, Israel. yosi.keller@gmail.com.

representation of each SP in terms of its color distributions. Succeeding video frames are modeled by a Markov Random Field (MRF), thus, casting the tracking as a maximum likelihood labeling problem, that is efficiently solved by spectral graph matching [9]. This results in a multi-cue plug-and-play tracking scheme that allows to fuse multiple tracking cues. Thus, we present the following contributions:

- First, we derive a novel probabilistic framework for fusing multiple unary and pairwise tracking cues, encoding spatial and temporal information. The probabilistic framework is of particular importance, as such a fusion requires a common probabilistic representation.

- Second, the different cues are jointly encoded in a graph based MRF and the inference is formulated via spectral relaxation with a corresponding probabilistic interpretation. While MRF-based formulations have been previously applied to tracking [25, 25], the spectral graph matching based formulation is simpler and faster.

- Third, the probabilistic interpretation of the spectral MRF solver provides statistical inference [9]. Namely, it allows to estimate the probabilities of an SP to belong to the foreground/background, rather than considering the MRF as a pure optimization problem. This allows to integrate the proposed scheme within other probabilistic schemes.

- Last, the proposed tracking framework is exemplified using a mixture of **five** unary and pairwise tracking cues, and we detail the use of additional tracking cues, implying that the scheme can be adapted to diverse tracking scenarios, thus providing a general plug-and-play framework for visual tracking.

This work is organized as follows. In Section 2 we survey previous results on tracking and image representation by SPs. The proposed object tracking framework is presented in Section 3, and the statistical inference is detailed in Section 4. We discuss the properties of the proposed schemes and suggest future enhancements in Section 5. Our approach is experimentally verified in Section 6, while concluding remarks are given in Section 7.

2 Related works

Object tracking has attracted significant research attention, resulting in a plethora of works whose thorough review is beyond the scope of this work. Thus, we present a survey of seminal approaches, and those related to the proposed scheme. As the proposed approach utilizes image representation by way of SPs, we review over-segmentation schemes in Section 2.1.

A gamut of visual tracking results are based on template matching, that is often the simplest to implement, but might overlook background information and be less adaptive to changes. Jepson *et al.* [13] proposed a stable appearance model, based on steerable pyramids, along with two-frame motion information. The stable component adapts to slowly varying properties of the image appearance by encoding properties that remain stable over long periods of time, while the two-frame motion constraints adapts to high frequency appearance changes.

Color histograms were shown to provide a useful and robust similarity measure due to their simplicity and invariance to geometrical appearance variations. In their seminal work Comaniciu *et al.* [6] introduced the Bhattacharyya coefficient as a color similarity measure, after smoothing the object with an isotropic Mean-Shift kernel. Wang *et al.* [28] developed a similarity measure based on color histograms, that utilizes the spatial layout of the colors, such that the object is represented by the joint probability distribution of spatial color information using GMMs. Such schemes represent the object as a region of interest and do not strive for pixelwise segmentation.

A common class of tracking algorithms is based on statistical inference in general, and classification in particular. Avidan *et al.* [2] proposed to iteratively update a set of weak classifiers to extract an object from its background. Each pixel is represented by a feature vector consisting of a local orientation histogram and the pixel's color. Once the weak classifiers are trained, a strong classifier is calculated using AdaBoost, producing a confidence map of the pixels, that is used to determine the new location of the object using Mean-Shift.

An online scheme for adaptive features selection for tracking was proposed by Collins *et al.* [5]. This discriminative learning approach seeks for the features that best differentiate the foreground and background utilizing RGB color histograms as features.

Semi-supervised learning was proposed as a mean of combining a fixed detector, as in classification and on-line learning methods. Grabner *et al.* [16] applied SemiBoost to learn such a classifier. The

weak classifiers are updated whenever a new training sample is available, and estimate a confidence map. Semi-supervised learning was used by Li *et al*. [19] as well, to combine object detection and tracking, by merging a trained detector with a two-frame template matching tracker.

The proposed scheme follows the same paradigm as Ren and Malik [21] who formulate the tracking as an iterative temporal foreground/background segmentation of an object from its background in each frame, by combining static image cues and temporal coherence models of the SPs in the image. The models of temporal coherence include appearance, scale, and spatial support, used to compute a mask by estimating the posterior marginal probabilities of the objects vs. background using a Conditional Random Field. A related approach was presented by Corrigan *et al*. [7], who suggested a user-free matting initialization by subtracting two sequential frames, followed by robust data modeling using Mean-Shift. Motion as well as color cues are used to form joint color and motion feature vectors. This matte allows GMMs to be trained for both the foreground and background, while being recursively refined using an iterative approach similar to GrabCut.

Criminisi *et al*. [8] presented a real-time approach to foreground/background separation, by probabilistically merging motion, color and contrast values alongside spatial and temporal priors. The inference is formulated via a Conditional Random Field model, that utilizes temporal and spatial gradients, and inferred by binary graph cuts.

In some recent works the tracking problem was formulated as a discrete optimization problem via graph based representations. Zha et al. [29] initialized the tracking manually, and new candidate positions were found using a Particle Filter. A graph is constructed, whose weights are computed using transductive learning. Leibe *et al*. [15] presented an approach for multi-object tracking by merging object detection and tracking in a hypothesis selection framework. They generate an over-complete set of hypothetical models for object detection and trajectory estimation, and the optimal subset is selected using the minimum description length (MDL) criterion.

The work of Sheikh and Shah [22] is of particular interest, as similar to our approach it utilizes both color domain modeling and discrete optimization via MAP-MRF. The scheme induces a spatial context into the color modeling, by computing non-parametric density estimates over the features space $\{R, G, B, x, y\}$, where $\{x, y\}$ are the pixels' coordinates. Further spatial context is achieved by defining an MRF model over adjacent pixels. This approach allows to learn the background model and thus detect

the motion of objects entering the scene, and then track them. Our approach can be thus viewed as its extension, as we reformulate the MRF inference using probabilistic spectral graph matching [9].

Tsai et al. presented an offline tracking scheme based on a multi-label MRF [25]. Their approach utilizes both motion and segmentation cues in a unified scheme, and enforces coherence within and across successive frames. The MRF is computed with respect to a volume of data, where each frame is a plane. Thus, the video sequence is analyzed as a whole, allowing to agglomerate tracking cues over the entire sequence. The downside of such an approach, is the significant increase in the dimensionality and computational complexity of the MRF inference problem.

Multiple-instance learning (MIL) is a supervised classification scheme, where a set of positive and negative *bags* is given, in contrast to labeled samples. A bag is labeled negative if all the samples in it are negative, and is labeled positive if it has at least one positive sample. Babenko *et al*. [3] applied the MIL approach to train a discriminative classifier for the 'tracking by detection' problems, in which the image is represented by a set of Haar-like features, that are computed for each image patch.

2.1 Superpixels representations

In this work we represent images by superpixels (SPs) that are a set of small homogenous image segments. The use of SPs is motivated by the fact that pixels are not perceptually meaningful entities, and are not scale invariant. In contrast, SPs allow to statistically represent image patches, either parametrically by Gaussians, or non-parametrically by histograms.

A common approach to the computation of SPs is the Watershed transform, where images are represented by topographic maps, and the gray level of a pixel stands for the elevation at that point. Watershed segmentation [23, 27] is a region-growing algorithm that partitions an image by simulating the process of water filling. The image appears as a topographic surface, that is flooded from its regional minima, while preventing the merging of the 'water' stemming from different sources.

The term SPs was coined in the seminal work of Ren and Malik *et al*. [20], that applied the normalized cut scheme to recursively partition an image using contours and texture features. To enforce locality, only local connections are taken into account when constructing the affinity matrix. Contrary to the Watershed scheme, all SPs extracted from an image have similar scale, making this method not scale-invariant. The TurboPixel algorithm [18] segments an image into a lattice-like structure of com-

pact SPs by dilating seeds. It produces segments that correspond to local image boundaries, and reduce under-segmentation through a compactness constraint. Given a user-specified value of K SPs, K seeds are placed in a radius of one pixel each, and the seeds grow in the direction of the image gradient as a function of the gradient magnitude.

Vedaldi and Soatto *et al*. [26] proposed the Quick Shift SP scheme, that is a mode seeking algorithm, that forms a tree of links to the nearest neighbor in order to increase the density. Unlike SP schemes based on normalized cuts, the SPs produced by Quick Shift are not of fixed size or number, as a complex image might have many more SPs than a simpler one. Therefore, there is no limitation on the boundary, leading to a varying size and shape of the detected SPs.

3 Video Tracking using Spectral Graph Matching

In this section we introduce the proposed tracking scheme. The gist of our approach is to derive a unified probabilistic formulation that fuses multiple heterogeneous tracking cues. The fusion of such cues requires a common representation provided by assignment probabilities, namely, the probabilities of image elements to belong to the object or background. For sake of clarity, we use five binary and unary cues, listed in Table 1, that were used in contemporary works [28, 3, 22], and that we found efficient. This choice is not unique, and one can choose other combinations of tracking cues, that would depend on the particular tracking scenario (static vs. moving camera for instance). We detail possible extensions in general, and the use of additional tracking cues in particular, in Section 5.

Each video frame $\mathbf{F}_t$ is represented by a set of SPs $\left\{S_i^t\right\}_{i=1}^{N_t} \in \mathbf{F}_t$. Thus, we utilize **five** cues consisting of two pairwise assignment probability cues. The first is based on the color similarity between SPs in the current frame $\mathbf{F}_t$ and is discussed in Section 3.1. The second, discussed in Section 3.2, relates SPs in $\mathbf{F}_t$ to overlapping SPs with known assignments in $\mathbf{F}_{t-1}$. We also utilize three unary cues, the first of which is detailed in Section 3.3, utilizing the GMM color models of the foreground $\mathcal{O}$ or background $\mathcal{B}$, computed using $\mathcal{O}_{t-1}$ and $\mathcal{B}_{t-1}$, the labeling of the tracked object and background in the preceding frame $\mathbf{F}_{t-1}$. We also use the Multiple Instance Learning (MIL) scheme in Section 3.4 to compute a unary term that relates $\mathcal{O}_{t-1}$ and $\mathcal{B}_{t-1}$ to SPs in $\mathbf{F}_t$, without computing a global GMM model as in Section 3.3. The third unary cue detailed in Section 3.5, utilizes a simple motion coherence scheme that induces constraints on the tracked object's motion. The different tracking cues are summarized in Table

1.

In order to fuse such heterogeneous probabilistic cues, we derive an inference scheme to compute the marginal probability of assigning each SP to either the $\mathcal{O}$ or $\mathcal{B}$. For that we encode the different tracking cues as probabilities in a graph corresponding to a Markov Random Field (MRF), and the probabilistic inference is computed via the spectral graph matching scheme presented in Section 4.

3.1 Pairwise color similarity

The pairwise affinities quantify the probability of two SPs to belong to the same assignment ($\mathcal{O}$ or $\mathcal{B}$), where similar SPs are expected to have the same assignments. For that, each frame is represented in the LAB color space, that was shown to strongly correlate with human color discrimination. Let $\mathbf{S}_i^t \in \mathbf{F}_t$ be a SP at frame t, and let $L_i^t = \{\mathcal{O}, \mathcal{B}\}$ be its (unknown) label. As the SPs correspond to homogeneous image patches, each SP is represented by a Gaussian G_i^t. The distance between neighboring SPs is computed by the Kullback-Liebler (KL) Divergence [14] between their corresponding Gaussians

$$D_{KL}(\mathbf{G}_i \| \mathbf{G}_j) = \frac{1}{2}\left(\log\left(\frac{|\mathbf{\Sigma}_j|}{|\mathbf{\Sigma_i}|}\right) + \mathrm{tr}\left(\mathbf{\Sigma}_j^{-1}\mathbf{\Sigma_i}\right) + (\mu_j - \mu_i)^\top \mathbf{\Sigma}_j^{-1}(\mu_j - \mu_i)\right), \quad (3.1)$$

where Σ_i and μ_i are the covariance matrix and mean vector, respectively, of G_i^t. We symmetrize Eq. 3.1 by

$$D_{KL}^s(\mathbf{S}_i \| \mathbf{S}_j) = \max(D_{KL}(\mathbf{S}_i \| \mathbf{S}_j), D_{KL}(\mathbf{S}_j \| \mathbf{S}_i)). \quad (3.2)$$

Thus, the pairwise assignment probabilities are defined using a Radial Basis Functions (RBF) kernel

$$\begin{aligned} P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^t \in \mathcal{O}\right) &= P\left(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_j^t \in \mathcal{B}\right) \\ &\propto \exp(-\frac{D_{KL}^s(G_i^t \| G_j^t)}{\sigma_p}) \end{aligned} \quad (3.3)$$

and

$$\begin{aligned} P\left(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_j^t \in \mathcal{O}\right) &= P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^t \in \mathcal{B}\right) \\ &\propto 1 - P\left(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_j^t \in \mathcal{B}\right), \end{aligned} \quad (3.4)$$

where σ_p is computed by

$$\sigma_p = \min\left(\underset{G_i^t, G_j^t \in \mathcal{O}}{median} D_{KL}^s(G_i^t \| G_j^t), \underset{G_i^t, G_j^t \in \mathcal{B}}{median} D_{KL}^s(G_i^t \| G_j^t)\right). \quad (3.5)$$

The two terms in Eq. 3.5 estimate the average KL divergence between SPs within $\mathcal{O}$ and $\mathcal{B}$, respectively, and the smaller one is chosen to better differentiate between the two clusters. σ_p is recomputed at each frame $\mathbf{F}_t$ based on the labeling of the SPs in the preceding frame.

3.2 Pairwise-constrained temporal similarities

In order to define the similarities between SPs in $\mathbf{F}_t$ and $\mathbf{F}_{t-1}$, we utilize the same pairwise probabilistic formulation as in Section 3.1, and apply it to SPs from $\mathbf{F}_t$ and $\mathbf{F}_{t-1}$, to induce temporal similarities. For each $\mathbf{S}_i^t \in \mathcal{O}_t \in \mathbf{F}_t$, we find the set $\mathbf{NN}_i^t = \left\{\mathbf{S}_{i_m}^{t-1}\right\}_{m=1}^{M_i} \in \mathbf{F}_{t-1}$ that is the set of SPs in $\mathbf{F}_{t-1}$ spatially overlapping with $\mathbf{S}_i^t$. We then compute pairwise assignment probabilities $P\left(\mathbf{S}_i^t, \mathbf{S}_j^{t-1}\right)$ $\forall \mathbf{S}_j^{t-1} \in \mathbf{NN}_i^t$ as in Eqs. 3.3 and 3.4. The constraint is given by the assignments of the SPs in $\mathbf{S}_j^{t-1} \in \mathbf{NN}_i^t$ that have already been inferred in the preceding frame $\mathbf{F}_{t-1}$, and is induced during the inference step discussed in Section 4.

3.3 GMM color models of the object and background

The unary probabilities $P_{GMM}\left(\mathbf{S}_i^t \in \mathcal{O}\right)$ and $P_{GMM}\left(\mathbf{S}_i^t \in \mathcal{B}\right)$ are modeled by GMM color models of the tracked object and background, GMM_O and GMM_B, respectively. The GMMs are estimated using an Estimation-Maximization (EM) scheme [4], where both GMMs are evaluated once in the first frame. This computation of GMM_O allows to avoid the drifting effect of the tracked object. The update procedure of the background color model GMM_B depends on the camera motion, for static camera (stationary background) GMM_B has to be updated less frequently, relating to illumination changes.

The distance between an SP $\mathbf{S}_i^t$ and the object or background is computed using the KL divergence between the corresponding Gaussian G_i^t and GMM. Unfortunately, there is no closed form expression for that KL divergence, and we apply the approximation proposed by Goldberger et al. [10],

$$\tilde{D}_{KL}(\mathbf{G}_i \| GMM_O) = \min_j (D_{KL}(\mathbf{G}_i \| o_j) - \log(\alpha_j)) \quad (3.6)$$

where

$$GMM_O = \mathbf{\Sigma}_{k=1}^{K} \alpha_k o_k, \tag{3.7}$$

K being the number of Gaussians in the GMM and α_k is the prior of a Gaussian o_k. The distance between SPs and GMM_B is calculated mutatis mutandis. The corresponding probabilities are then given by

$$P_{GMM}\left(\mathbf{S}_i^t \in L_i^t\right) \propto \exp\left(-\frac{\tilde{D}_{KL}(\mathbf{G}_i^t \| GMM_{L_i^t})}{\sigma_u}\right), \tag{3.8}$$

where σ_u is computed similarly to Eq. 3.5

$$\sigma_u = \min\left(\underset{G_i^t \in \mathcal{O}}{median} \tilde{D}_{KL}(\mathbf{G}_i^t \| GMM_{\mathcal{O}}), \underset{G_i^t \in \mathcal{B}}{median} \tilde{D}_{KL}(\mathbf{G}_i^t \| GMM_{\mathcal{B}})\right). \tag{3.9}$$

3.4 Inducing temporal coherence via Multiple Instance Learning

Multiple Instance Learning (MIL) [3] is a supervised learning approach to inference problems with inconclusive labeling of the training samples. Namely, the labeling of the training samples might be erroneous. In the MIL framework, the samples are assigned to bags of instances, where in the binary labeling case, a bag is labeled positive if at least *one* instance in that bag is positive, and the bag is labeled negative if *all* of its instances are negative. The goal of MIL is to label unseen bags or instances based on the labeled bags as the training data.

In the context of proposed tracking scheme, we assume that most SPs of interest are correctly labeled as $\mathcal{O}$ in $\mathbf{F}_{t-1}$. But, there might be a few outlier background SPs, that are wrongly labeled as $\mathcal{O}$. Hence we aim to apply the MIL approach that is able to overcome such obstacles.

We denote the set of object SPs labeled at time $t-1$ as $\mathcal{O}_{t-1}$ and consider it a bag

$$\mathcal{O}_{t-1} = \begin{cases} \mathcal{O} \text{ (positive)} & \exists j \text{ s.t. } \mathbf{S}_j^{t-1} \in \mathcal{O} \\ \mathcal{B} \text{ (negative)} & \mathbf{S}_j^{t-1} \in \mathcal{B}\ \forall j \end{cases}, \tag{3.10}$$

where the foreground $\mathcal{O}$ is the positive label and $\mathcal{B}$ is the negative one. Let L_i^t be the label of $\mathbf{S}_j^t$, then

$$\begin{aligned} L_i^t &= \arg\max_{L_i^t \in \{\mathcal{O},\mathcal{B}\}} P(\mathbf{S}_i^t \in L_i^t | \mathcal{O}_{t-1}, \mathcal{B}_{t-1}) \\ &= \arg\max_{L_i^t \in \{\mathcal{O},\mathcal{B}\}} P(\mathbf{S}_i^t \in L_i^t | \mathcal{O}_{t-1}) P(\mathbf{S}_i^t \in L_i^t | \mathcal{B}_{t-1}) \end{aligned} \tag{3.11}$$

where we assumed that the bags are conditionally independent.

The positive set $\mathcal{O}_{t-1}$ is given by the tracking result of $\mathbf{F}_{t-1}$, while the negative set $\mathcal{B}_{t-1}$ is a narrow image strip around $\mathcal{O}_{t-1}$, computed by dilating the binary mask corresponding to $\mathcal{O}_{t-1}$, as depicted in Fig. 1b. $P(\mathbf{S}_i^t \in L_i^t|\mathcal{O}_{t-1})$ follows the definition of the positive set $\mathcal{O}$

$$P(\mathbf{S}_i^t \in L_i^t|\mathcal{O}_{t-1}) = 1 - \prod_j \left(1 - P(\mathbf{S}_i^t \in L_i^t|\mathcal{O}_{t-1}^j)\right). \tag{3.12}$$

and

$$\begin{cases} P(\mathbf{S}_i^t \in \mathcal{B}|\mathcal{O}_{t-1}) = 1 - \prod_j P(\mathbf{S}_i^t \in \mathcal{O}|\mathcal{O}_{t-1}^j) \\ P(\mathbf{S}_i^t \in \mathcal{O}|\mathcal{O}_{t-1}) = 1 - \prod_j \left(1 - P(\mathbf{S}_i^t \in \mathcal{O}|\mathcal{O}_{t-1}^j)\right) \end{cases}. \tag{3.13}$$

Similarly,

$$P(\mathbf{S}_i^t \in L_i^t|\mathcal{B}_{t-1}) = \prod_j P(\mathbf{S}_i^t \in L_i^t|\mathcal{B}_{t-1}^j). \tag{3.14}$$

and

$$\begin{cases} P(\mathbf{S}_i^t \in \mathcal{B}|\mathcal{B}_{t-1}) = \prod_j P(\mathbf{S}_i^t \in \mathcal{B}|\mathcal{B}_{t-1}^j) \\ P(\mathbf{S}_i^t \in \mathcal{O}|\mathcal{B}_{t-1}) = \prod_j \left(1 - P(\mathbf{S}_i^t \in \mathcal{B}|\mathcal{B}_{t-1}^j)\right) \end{cases}. \tag{3.15}$$

The probabilities $P(\mathbf{S}_i^t \in \mathcal{O}_t|\mathcal{O}_{t-1}^j)$ and $P(\mathbf{S}_i^t \in \mathcal{B}_t|\mathcal{B}_{t-1}^j)$ are approximated using the RBF kernel

$$\begin{aligned} P(\mathbf{S}_i^t \in \mathcal{O}_t|\mathcal{O}_{t-1}^j) &= P(\mathbf{S}_i^t \in \mathcal{B}_t|\mathcal{B}_{t-1}^j) \\ &\propto \exp\left(-\frac{D_{KL}(\mathbf{S}_i^t \| \mathbf{S}_j^{t-1})}{\sigma_p}\right), \end{aligned} \tag{3.16}$$

where σ_p is computed as in Eq. 3.5, and we denote the MIL based labeling probability as

$$P_T\left(\mathbf{S}_i^t \in L_i^t\right) = P(\mathbf{S}_i^t \in L_i^t|\mathcal{O}_{t-1}) P(\mathbf{S}_i^t \in L_i^t|\mathcal{B}_{t-1}). \tag{3.17}$$

3.5 Encoding hard tracking constraints

At each time instance t we are given $\mathcal{O}_{t-1}$, an estimate of the object's footprint in $\mathbf{F}_{t-1}$. As the motion of the tracked object is limited by physical constraints such as maximal velocity and acceleration, $\mathcal{O}_{t-1}$ and $\mathcal{O}_t$ are known to overlap, as depicted in Fig. 1. Thus, the SPs in $\mathbf{F}_t$ that overlap with $\mathcal{O}_{t-1}$, are apriori known to belong to $\mathcal{O}_t$. These constraints can be induced on the tracking results. To estimate the overlap, it is common to estimate the object's motion over time using a Kalman Filter. For sake of clarity, we implemented a simpler derivation, where we assumed a maximal object velocity and computed the spatial overlap between $\mathcal{O}_{t-1}$ and $\mathcal{O}_t$ by morphologically eroding $\mathcal{O}_{t-1}$.

(a) (b) (c)

Figure 1: Hard tracking constraints. Given the tracking result of $\mathbf{F}_{t-1}$ in (a), we apply erosion to derive the area (marked in white) in $\mathbf{F}_t$ (b), that can be assumed to belong to the tracked object. By applying morphological dilation we derive the outer gray area that is used to model the background. (c) An estimate of the overlap of $\mathbf{F}_{t-1}$ and $\mathbf{F}_t$.

4 Statistical inference via spectral relaxation

Given the pairwise and unary assignment probabilities computed in the previous section, we aim to derive a unified probabilistic formulation and apply a corresponding inference scheme. We represent the different probabilistic cues via a graph, or equivalently an MRF. The different tracking cues are summarized in Table 1

Cue	Type	Notation	Section
Pairwise color similarity	binary	$P\left(S_i^t \in L_i^t, S_j^t \in L_j^t\right)$	3.1
Temporal pairwise color similarity	binary	$P\left(S_i^t \in L_i^t, S_j^{t-1} \in L_j^t\right)$	3.2
Foreground and background color modeling	unary	$P_{GMM}\left(S_i^t \in L_i^t\right)$	3.3
MIL-based temporal coherence	unary	$P_T\left(S_i^t \in L_i^t\right)$	3.4
Hard tracking constraints	unary	–	3.5

Table 1: The tracking cues used in the proposed scheme.

The unary probabilities are computed using the GMM ($P_{GMM}\left(\mathbf{S}_i^t \in L_i^t\right)$) and MIL-based $P_T\left(\mathbf{S}_i^t \in L_i^t\right)$ probabilities defined in Eqs. 3.8 and 3.13, respectively. We assume the probabilities are *conditionally* independent as $P_{GMM}\left(\mathbf{S}_i^t \in L_i^t\right)$ and $P_T\left(\mathbf{S}_i^t \in L_i^t\right)$ are essentially conditional probabilities, conditioned on their different cues and estimation schemes. Namely, the estimation of the probability $P\left(\mathbf{S}_i^t \in L_i^t\right)$ using a global GMM model ($P_{GMM}\left(\mathbf{S}_i^t \in L_i^t\right)$) or the MIL model ($P_T\left(\mathbf{S}_i^t \in L_i^t\right)$ results in independent

estimates

$$P_u\left(\mathbf{S}_i^t \in L_i^t\right) = P_{GMM}\left(\mathbf{S}_i^t \in L_i^t\right) P_T\left(\mathbf{S}_i^t \in L_i^t\right), \tag{4.1}$$

encoded by the unary potential

$$\Psi_1\left(\mathbf{S}_i^t\right) = P_u\left(\mathbf{S}_i^t \in L_i^t\right), \;\; i = 1..N_t. \tag{4.2}$$

The pairwise potential is based on the pairwise probabilities defined in Eqs. 3.3 and 3.4

$$\Psi_2\left(\mathbf{S}_i^t, \mathbf{S}_j^t\right) = \begin{cases} P\left(\mathbf{S}_i^t \in L_i^t, \mathbf{S}_j^t \in L_j^t\right) & \mathbf{S}_i^t \text{ and } \mathbf{S}_j^t \text{ are spatially adjacent} \\ 0 & \text{otherwise} \end{cases}. \tag{4.3}$$

The constrained-pairwise probabilities relate spatially overlapping SPs in $\mathbf{F}_t$ and $\mathbf{F}_{t-1}$

$$\widehat{\Psi}_2\left(\mathbf{S}_i^t, \mathbf{S}_j^{t-1}\right) = \begin{cases} P\left(\mathbf{S}_i^t \in L_i^t, \mathbf{S}_j^{t-1} \in L_j^{t-1}\right) & \mathbf{S}_j^{t-1} \in \mathbf{NN}_i^t \\ 0 & \text{otherwise} \end{cases}, \tag{4.4}$$

where $\mathbf{S}_j^{t-1} \in \mathbf{NN}_i^t$ are the SPs in $\mathbf{F}_{t-1}$ that spatially overlap with $\mathbf{S}_i^t$.

Let $\mathbf{X} \in \{0,1\}^{(N_t+N_{t+1})\times 2}$ be an assignment matrix, such that $x_{i,1}+x_{i,2} = 1, \forall i = 1..\left(N_t + N_{t+1}\right)$, implying that the i'th row of $\mathbf{X}$ encodes the assignment of $\mathbf{S}_i$. Namely, $x_{i,1} = 1$ (and $x_{i,2} = 0$) implies that $\mathbf{S}_i \in \mathcal{O}$, and viceversa. $\mathbf{x} \in \{0,1\}^{2(N_t+N_{t+1})}$ is a rowwise vectorized replica of $\mathbf{X}$, that is commonly used in the formulation of assignment problems.

Thus, the tracking task can be cast as the following optimization problem

$$\begin{aligned} \mathbf{x}^* = \arg\max_{\mathbf{x}} & \sum_i x_i P_u\left(\mathbf{S}_i^t\right) \\ & + \alpha^2 \sum_{i,j} x_i x_j P\left(\mathbf{S}_i^t, \mathbf{S}_j^t\right) + \beta^2 \sum_{i,j} x_i x_j P\left(\mathbf{S}_i^t, \mathbf{S}_j^{t-1}\right) \\ & s.t.\ \mathbf{x} \in \{0,1\}^{2(N_t+N_{t+1})} \text{ and} \\ & \mathbf{x}((2N_t+1) : 2\left(N_t + N_{t+1}\right)) = \mathbf{x}_{t-1}\left(1 : 2N_{t-1}\right) \end{aligned} \tag{4.5}$$

where $\mathbf{x}_{t-1}$ is the assignment vector computed for the preceding frame $\mathbf{F}_{t-1}$, and α and β are predefined weights that control the trade-off between the unary and binary terms.

Equation 4.5 can be reformulated as a quadratic assignment problem (QAP), such that

$$\begin{aligned}\mathbf{x}^* &= \arg\max_{\mathbf{x}} \left(\mathbf{x}^T\mathbf{c} + \alpha^2 \mathbf{x}^T \mathbf{A}\mathbf{x} + \beta^2 \mathbf{x}^T \widehat{\mathbf{A}}\mathbf{x} \right) \\ &= \arg\max_{\mathbf{x}} \left(\mathbf{x}^T \left(\alpha^2 \mathbf{A} + \beta^2 \widehat{\mathbf{A}} + diag\left(\mathbf{c}\right) \right) \mathbf{x} \right) \\ &= \arg\max_{\mathbf{x}} \left(\mathbf{x}^T \tilde{\mathbf{A}}\mathbf{x} \right) \\ &s.t.\ \mathbf{x}\left((2N_t+1) : 2\left(N_{t-1}+N_t\right)\right) = \\ &\qquad \mathbf{x}_{t-1}\left(1 : 2N_{t-1}\right),\ \mathbf{x} \in \{0,1\}^{2(N_t+N_{t-1})} \end{aligned} \tag{4.6}$$

where $\mathbf{c}, \mathbf{A}$, and $\widehat{\mathbf{A}}$ are the matrix representation of the corresponding unary and binary terms in Eq. 4.5, and $diag\left(\mathbf{c}\right)$ is a diagonal matrix whose main diagonal is the vector $\mathbf{c}$. The constraint in Eq. 4.6 implies that the assignment of the SPs in $\mathbf{F}_{t-1}$ are used as hard constraints. The matrix $\hat{\mathbf{A}} \in \mathbb{R}^{2(N_t+N_{t-1})\times 2(N_t+N_{t-1})}$ consists of

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A}_{t,t} & \mathbf{A}_{t,t-1} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.7}$$

where $\mathbf{A}_{t,t} \in \mathbb{R}^{2N_t \times 2N_t}$ consists of 2×2 blocks $\mathbf{C}_{ij}$. The off-diagonal blocks encode the pairwise potentials $\Psi_2\left(\mathbf{S}_i^t, \mathbf{S}_j^t\right)$ such that

$$\mathbf{C}_{i,j} = \begin{pmatrix} P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^t \in \mathcal{O}\right) & P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^t \in \mathcal{B}\right) \\ P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^t \in \mathcal{B}\right) & P\left(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_j^t \in \mathcal{B}\right) \end{pmatrix}, \quad \forall i, j = 1..N_t, i \neq j \tag{4.8}$$

and the 2×2 blocks on the main diagonal encode the unary potentials $\Psi_1\left(\mathbf{S}_i^t\right)$ such that

$$\mathbf{C}_{i,j} = \begin{pmatrix} P_u\left(\mathbf{S}_i^t \in \mathcal{O}\right) & 0 \\ 0 & P_u\left(\mathbf{S}_i^t \in \mathcal{B}\right) \end{pmatrix}, \ \forall i = 1..N_t, \tag{4.9}$$

The constrained-pairwise probabilities are encoded in $\mathbf{A}_{t,t-1} \in \mathbb{R}^{2N_{t-1}\times 2N_{t-1}}$ in 2×2 blocks such that

$$\mathbf{C}_{i,N_t+j} = \begin{pmatrix} P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^{t-1} \in \mathcal{O}\right) & P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^{t-1} \in \mathcal{B}\right) \\ P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_j^{t-1} \in \mathcal{B}\right) & P\left(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_j^{t-1} \in \mathcal{B}\right) \end{pmatrix}, \quad i = 1..N_t, j = 1..N_{t-1}, i \neq j \tag{4.10}$$

where the diagonal 2×2 blocks are zeroed.

As each SP is assigned to a single label, the pairwise and unary probabilities adhere to

$$P(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_i^t \in \mathcal{O}) + P\left(\mathbf{S}_i^t \in \mathcal{O}, \mathbf{S}_i^t \in \mathcal{B}\right) + P(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_i^t \in \mathcal{O}) + P(\mathbf{S}_i^t \in \mathcal{B}, \mathbf{S}_i^t \in \mathcal{B}) = 1 \quad (4.11)$$

and

$$P(\mathbf{S}_i^t \in \mathcal{O}) + P\left(\mathbf{S}_i^t \in \mathcal{B}\right) = 1, \quad (4.12)$$

and $\hat{\mathbf{A}}$ is normalized accordingly.

The binary quadratic optimization problem in Eq. 4.6 corresponds to a two-dimensional MRF, and is known to be NP-hard. Leordeanu and Hebert *et al.* [17] proposed an efficient and robust solution based on spectral relaxation, where Eq. 4.6 is relaxed to

$$\mathbf{w}^* = \arg\max_{\mathbf{w}} \frac{\mathbf{w}^T \tilde{\mathbf{A}} \mathbf{w}}{\mathbf{w}^T \mathbf{w}} \ , \ \mathbf{w} \in \mathbb{R}^{2(N_t + N_{t-1})} \quad (4.13)$$

Equation 4.13 is a Rayleigh quotient and can thus be solved by $\mathbf{w}^*$ being the principal eigenvector of $\tilde{\mathbf{A}}$. As $\tilde{\mathbf{A}}$ is nonnegative and symmetric, by the Perron-Frobenius theorem we have that its principal eigenvector is known to exist, and is nonnegative. Egozi and Keller [9] showed that if the different assignments are statistically independent, and the affinity matrix $\tilde{\mathbf{A}}$ is an empirical estimate of the joint assignment probability, this scheme boils down to an estimate of the marginal assignment probabilities.

As we compute the principal eigenvector of $\tilde{\mathbf{A}}$, we apply a Constrained Power Iteration algorithm, which is a variation of the common Power Iteration [11], that is modified to induce the tracking constraints discussed in Section 3.5. Let $o = \{o_1, ..., o_O\}$ be the indexes of the SPs that are apriori known to belong to $\mathcal{O}_t$, the corresponding constraints to each such entry o_k can be encoded by

$$\mathbf{w}^* (2o_k - 1) = 1 \quad (4.14)$$
$$\mathbf{w}^* (2o_k) = 0.$$

The Power Iteration [11] is an iterative algorithm that solves a convex optimization problem, implying that each of its iterations is a projection of $\mathbf{w}^*$ on a convex set. The assignment in Eq. 4.14 also constitutes a convex projection, and we get that the solution vector $\mathbf{w}^*$ of the modified scheme converges to the intersection of both convex domains. This approach is summarized in Algorithm 1.

Algorithm 1 Computing the constrained principal eigenvector

Input: $\tilde{\mathbf{A}}$ The affinity matrix, and the number of iterations K

Output: $\mathbf{w}_K =$ the principal eigenvector

$\mathbf{w}_0 = U[0,1]^{2(N_t+N_{t+1})}$

for $k = 1 \rightarrow K$ **do**

$\mathbf{w}_k = \tilde{\mathbf{A}}\mathbf{w}_{k-1}$

Induce hard constraints on $\mathbf{w}_k$

$\mathbf{w}_{k-1} = \mathbf{w}_k$

end for

Given the continuous solution $\mathbf{w}^*$, it is discretized by choosing the assignment (per SP) with the maximal marginal assignment probability, to yield the discrete solution $\mathbf{x}^*$. As there are no assignment constraints, this greedy discretization is the Maximum Likelihood discretization.

5 Discussion and Future extensions

The proposed scheme can be related to contemporary state-of-the-at results. The seminal Mean-Shift [6] approach tracks a single region of interest by representing it by non-parametric statistics. It does not extract the contour of the object, as our scheme. Thus, some background pixels might be used for object modeling, resulting in a drifting effect, as exemplified in Section 6. Moreover, there is no use of additional cues, or background modeling. In the context of the proposed scheme, the Mean-Shift is a color domain similarity measure, that can be used to compute both pairwise (Section 3.1) and unary (Section 3.3) SPs similarities. MRF-based approaches, such as Sheikh et al. [22] and Tsai [25], provide improved performance over Mean-Shift. Our scheme provides an efficient MRF solver based on a probabilistic formulation of spectral graph matching [9], that allows the synergy of multiple cues, and while five were implemented in this work, more can be introduced as discussed in Section 5.3.

5.1 Motion detection

The initialization phase of our scheme is based on a given motion detection scheme, where in this work we used a bounding box manually marked by the user. Similar to Sheikh and Shah's work [22], the

proposed scheme can be extended to detect new objects entering the observed scene. This follows from the probabilistic inference scheme discussed in Section 4, where we compute the probability of a SP to belong to either $\mathcal{O}$ or $\mathcal{B}$. In the detection phase, the scheme will be applied as in Section 3, while omitting the object unary term, discussed in Section 3.3. The detection is given by computing the marginal SP probability $P\left(\mathbf{S}_i^t \in \mathcal{B}\right) < T$, where T is a predefined threshold.

5.2 Tracking multiple objects

The proposed scheme can be adapted to track multiple objects. First, multiple objects can be tracked as parts of a single foreground entity, as our approach does not require spatial continuity of the tracked objects. This is exemplified in Figs. 8 and 9, where each object can be individually extracted using connected component analysis. Second, one can run the tracker multiple times, each instance tracking a particular object. The third extension is by formulating the inference scheme in Section 4, as a multilabel graph matching problem. The probabilistic graph matching scheme [9] can be extended to handle multiple labels, (typically up to 15). Hence, by computing the pairwise assignment probabilities between the tracked objects, and their unary potentials, we can solve the inference problem with respect to multiple labels. By using a detection scheme, all of these three multi-object tracking schemes, can be applied dynamically by changing the number of tracked objects on the fly, from frame to frame.

5.3 Additional tracking cues

The proposed scheme can be further improved by utilizing additional tracking cues. Each of these cues can be trained separately and so is their probabilistic fusion. A thorough example of fusing and training multiple low-level vision cues was given by Alpert et al. [1] in the context of image segmentation.

5.3.1 GMM-based Change detection

Change detection is a powerful unary cue that is suitable for tracking using a stationary camera. Stauffer and Grimson [24] proposed a straightforward and efficient approach that models background pixels by GMMs, allowing to robustly handle time varying backgrounds due to swinging bushes, waving flags, atmospheric turbulence etc. These probabilities can then be agglomerated into the unary term $P_u\left(\mathbf{S}_i^t \in L_i^t\right)$ using Eq. 4.1. The potential use of this cue is exemplified in Section 6 in Figs. 11 and 12,

where we show tracking failures of the proposed scheme that could have been avoided.

5.3.2 Encoding hard tracking constraints using a Kalman filter

The hard tracking constraints discussed in Section 3.5, can be extended using a Kalman filter [4], where the object overlap in succeeding frames can be better estimated, and encoded the same as in Section 3.5. Moreover, the Kalman filter allows to compute the probabilities for each SPs and pixel to overlap with $\mathcal{O}_{t-1}$ in $\mathbf{F}_{t-1}$. Thus, this constraint can be used as unary assignment probabilities, same as in Eq. 4.1.

6 Experimental Results

We experimentally verified the proposed scheme by applying it to video sequences used in contemporary works. For that we used video sequences kindly provided by Sheikh and Shah [22][1], the CAVIAR project[2], the i-Lids dataset for AVSS 2007[3], the PETS 2001 dataset [4], and some of the sequences used in CVPR 2012 Change Detection Workshop [12][5]. Some of these sequences are given with groundtruth, allowing to assess the tracking accuracy at a pixel level. To simulate the common video format in surveillance systems, each video frame was resized to $\sim$ 1M pixels, and the object of interest was manually marked in the first frame by marking a bounding box. We used the same set of parameters in all of our experiments, where the number of mixture components in each GMM was $K = 3$, the RBF bandwidths were computed according to Eqs. 3.5 and 3.9, and the mixing weights in Eq. 4.6 were $\alpha = 1$ and $\beta = 1$. In the video sequences we analyzed, typically consisting of $o\left(100\right)$ frames, the background remained stationary and we found no gain in updating (or reestimating) the background model GMM GMM_B. We compare against the MRF-based scheme of Sheikh and Shah [22] whose implementation is publicly given [6], and the seminal Mean-Shift algorithm, implemented by Ning et al. [7].

Our approach is based on representing the video frames by SPs. For that we applied the Watershed

[1] Available at: http://www.cs.cmu.edu/~yaser/

[2] Videos taken from: http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/

[3] Available at: http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html

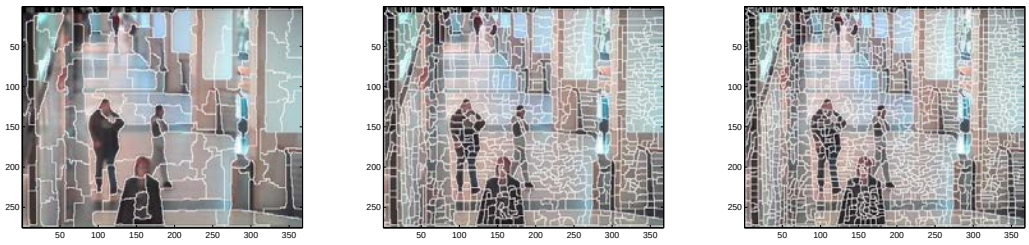
[4] Available at: http://www.anc.ed.ac.uk/demos/tracker/pets2001.html

[5] Available at: http://www.changedetection.net/

[6] Available at: http://www.cs.cmu.edu/~yaser/

[7] Available at: http://www4.comp.polyu.edu.hk/~cslzhang/CBWH.htm

algorithm with a threshold of $\tau = 0.7$ in all of our simulations. Figure 2 depicts the SPs achieved by using different values of threshold τ. Using $\tau = 0.25$ as in Fig. 2a results in coarse SPs that might contain both object and foreground components, while setting $\tau = 1.0$ results in over-segmentation, with small SPs whose corresponding Gaussians might have degenerate covariance matrices.



(a) $\tau = 0.25$ (b) $\tau = 0.7$ (c) $\tau = 1.0$

Figure 2: The superpixels produced by the Watershed over-segmentation algorithm for different values of the threshold τ.

In the early stages of our research we used normalized correlation to derive the temporal cue, instead of the MIL-based and constrained-pairwise terms discussed in Sections 3.4 and 3.2, respectively. This resulted in a drifting error due to appearance changes, as depicted in Fig. 3 for the sequence shown in Fig. 4, where the template matching scheme lost track of the limbs of the tracked subject, as their appearance varies due to the subject's walk. In contrast, in Fig. 4 we report the results of using the proposed scheme where the temporal cues are encoded via statistical measures, resulting in improved tracking without drifting errors.

(a) (b) (c) (d) (e) (f)

Figure 3: The drift error caused by a template matching based tracker. Once the tracker looses track of the limbs due to their appearance variations, it can not regain track. The sequence is depicted in Fig. 4.

6.1 Qualitative Analysis

Comparing different tracking schemes is an intricate task, as different approaches might relate to different variations of the problem. For instance, the Mean-Shift algorithm [6] and its variations track a bounding box enclosing the object of interest, while Sheikh and Shah [22], Tsai [25] as well as the proposed scheme, aim to extract the object's contour. Therefore, to compare against the Mean-Shift scheme we depict qualitative tracking comparisons. The method proposed by Sheikh et al. [22] uses a background learning phase of 200 frames, while Mean-Shift and the proposed scheme used the same initialization, given manually as a bounding box.

We first analyze a video sequence depicting a subject crossing a corridor [8] that is showed in Fig. 4, where we compare against the Mean-Shift and Sheikh et al. The tracked person is observed from different orientations, as his pose changes and the movement stops for a few seconds, partially hidden from the field of view. The proposed scheme is able to continuously track the object and extract its contour with high accuracy. In contrast, Mean-Shift, looses track due to the occlusion. Sheikh's algorithm shows better results (than the Mean-Shift), but its accuracy is inferior to the proposed scheme.

In the second experiment we analyzed the outdoor sequence shown in Fig. 5. This scene is characterized by appearance changes due to both geometrical changes (as in Fig. 4), and variations in lighting conditions. The proposed tracker manages to track the pedestrian through the entire sequence. Note that in the first frame the tracker failed to identify the pedestrian's head, but it manages to track his entire body throughout the sequence. Although the size of the object changed significantly throughout the sequence, it does not affect the tracking results. The Mean-Shift scheme was able to track the object correctly, but it is unable to estimate the spatial support of the tracked person.

The results shown in Figs. 6 and 7, relate to video sequences where subjects enter or leave the scene[9]. Thus, these scenes are characterized by significant dynamic changes in the pose and size of the tracked subject. Namely, the subject in the earlier frames is only partially overlapping with his manifestation in later frames. It follows that the proposed tracker manages to track a person through the entire sequence, and correctly extract the object of interest. The Mean-Shift scheme follows the person well in both figures, but is unable to adapt to the varying object size. Sheikh's algorithm was applied in Fig. 7 and

[8] Videos taken from: http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1/

[9] Available at: http://www.eecs.qmul.ac.uk/~andrea/avss2007_d.html

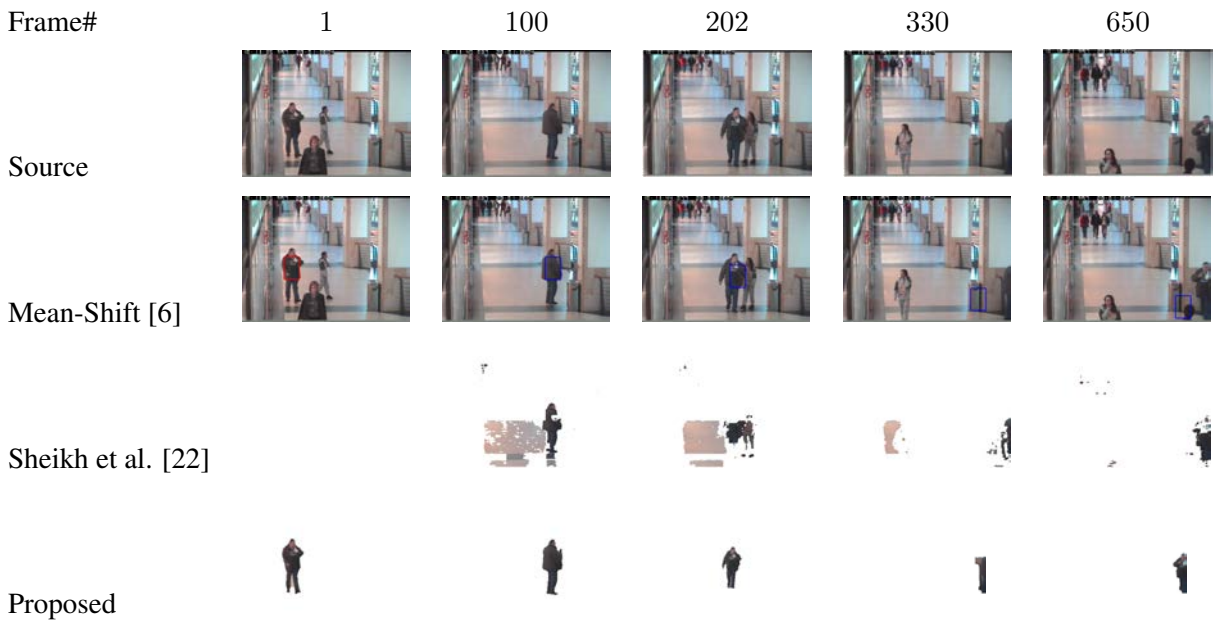


Figure 4: Indoor scene tracking results. The top row. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box. The results of Sheikh et al. [22] are delayed due to the background learning phase.

seems to lock on to the background, rather than the object of interest.

The tracking of multiple objects having colors similar to those of the background color is shown in Figs. 8 and 9. As the unary and pairwise similarly measures used by our approach are based on color cues, this is a challenging scenario. It follows that the proposed scheme is able to robustly track the subject, it is also able to track two object simultaneity, without using a particular setup. As before, the Mean-Shift scheme is able to track the objects, but fails to localize them, while Sheikh's algorithm lock on to the background. The tracking of the multiple objects is achieved automatically, as the proposed MRF formulation allows the tracked object to be spatially discontinues. Each tracked object can then be individually extracted using connected component analysis. Similar results are reported in Fig. 10, where we track an object undergoing a sudden motion. The results shown in Fig. 11 exemplify the proposed trackers' ability to track an object undergoing velocity and orientation changes.

The downside of using a global GMM color model is shown in Fig. 12, where our tracker failed to track a car crossing another car with a similar color[10]. This could have been resolved by using the

[10]Available at: http://www.anc.ed.ac.uk/demos/tracker/pets2001.html

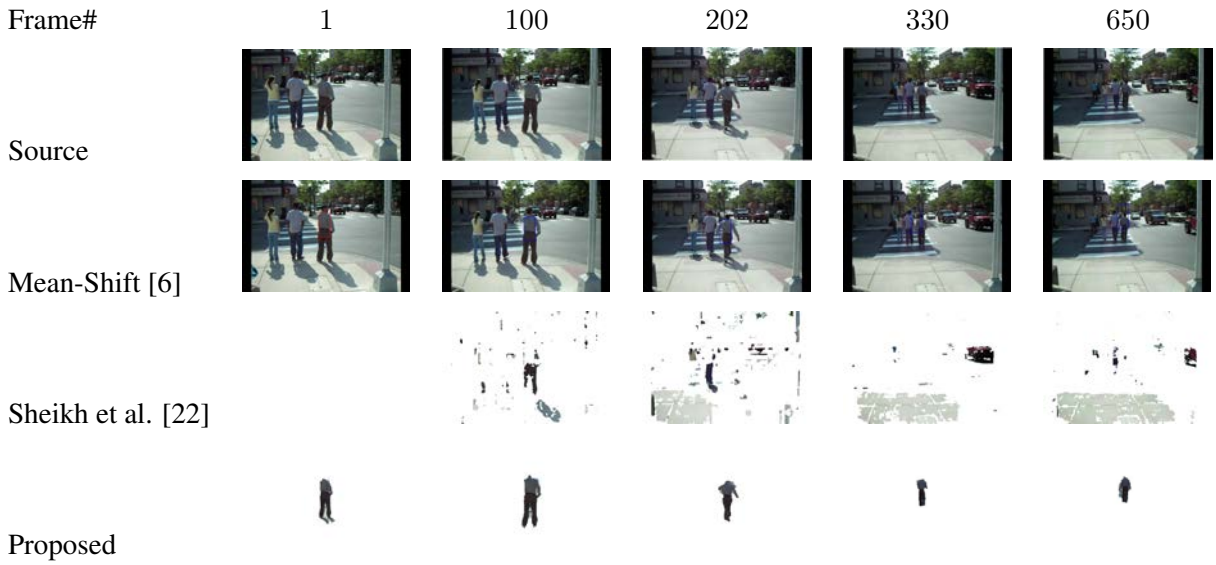


Figure 5: Tracking results of an object in an outdoor scene characterized by significant lighting, pose and size changes of the tracked subject. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

change detection cue discussed in Section 5.3.1, as this cue can differentiate between static and moving objects, especially in video sequences acquired by a static camera as in Figs. 11 and 12.

6.2 Quantitative Analysis

In order to quantify the performance of the proposed scheme, we applied the proposed schemes to sequences with given groundtruth segmentations. For that we used the sequences given in the CVPR 2012 Change Detection Workshop [12]. We compared against the scheme of Sheikh and Shah [22] that provides per pixel segmentation for each frame. We report the tracking accuracy in Figs. 13-15, in terms of Recall, Precision and Accuracy, such that

$$Recall = \frac{TP}{TP + FN} \tag{6.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{6.2}$$

$$Accuracy = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}. \tag{6.3}$$

where TP, FN and FP are the True Positives, False Negatives and False Positives, respectively. The Recall quantifies the ratio of labeled object pixels to those present in the groundtruth object. The Precision is the ratio of the number of *correctly labeled* object pixels to the number of those labeled as object

Figure 6: Tracking results for an object leaving a scene. The sequence is characterized by a significant change in the object's size, and it color GMM model. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

pixels. The Accuracy (F- measure) provides a unified precision measure in the range $[0, 1]$.

The tracking results of the *Highway* sequence are reported in Fig. 13. The high precision shown in Fig. 13b implies that Sheikh and Shah's scheme labeled object pixels with high certainty, but it only detected a small fraction of the object, manifested by the low Recall rate in Fig. 13c. In contrast, the proposed scheme is able to label most of the tracked object with high accuracy. It follows that the F-measure (in Fig. 13a) of our scheme is significantly higher. Similar results were achieved for the *Office* and *Pets2006* sequences in Figs. 14 and 15, respectively, where the proposed scheme labeled the object almost completely, as quantified by the Recall values. In contrast, Sheikh and Shah detected only a few pixels from the object, hence the low Recall and Accuracy measures. We also applied the two schemes to of Sheikh's groundtruth sequence [11], and the results are reported in Fig. 16, where Sheikh's approach outperformed the proposed scheme.

[11] Available at: http://www.cs.cmu.edu/~yaser/

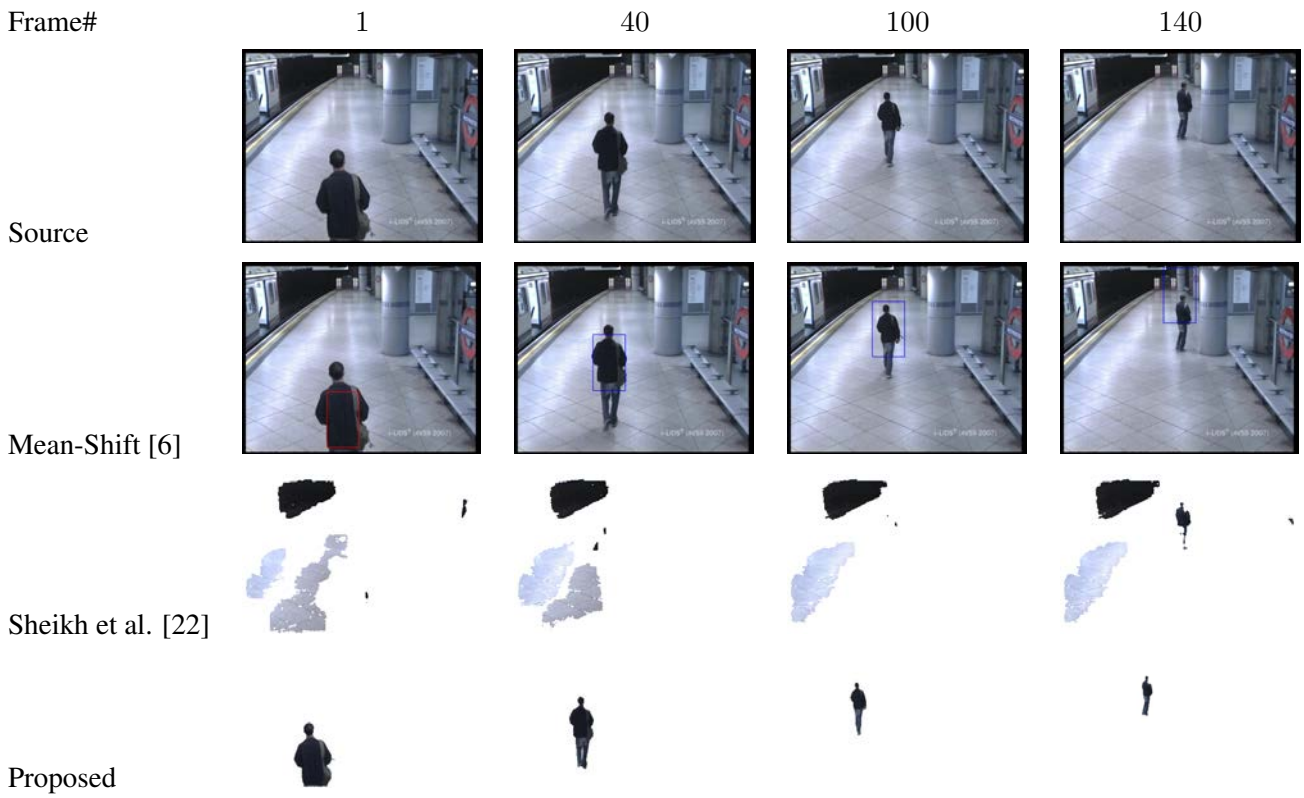


Figure 7: Tracking results for an object entering a scene. The sequence is characterized by a significant change in the object's size. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

6.3 Implementation issues and conclusions

The proposed scheme was implemented in Matlab, using its built-in implementations of GMM, and SP construction via the Watershed transform. Each video frame consists 100-200 SPs, where given the different tracking cues, the inference step based on spectral graph matching, requires 100ms on average. The analysis of each frame lasts 2 seconds, where the bulk of the running time is the computation of the tracking cues. As these can be computed in parallel, we believe that the proposed approach can be implemented as a real-time system.

To conclude, the proposed scheme was shown to compare favorably with the Means Shift and Sheikh and Shah's schemes on most (all but one) test sequences we evaluated. Compared to Means Shift, our scheme allows to extract the object's contour, that can be used for higher level analysis such as pose and

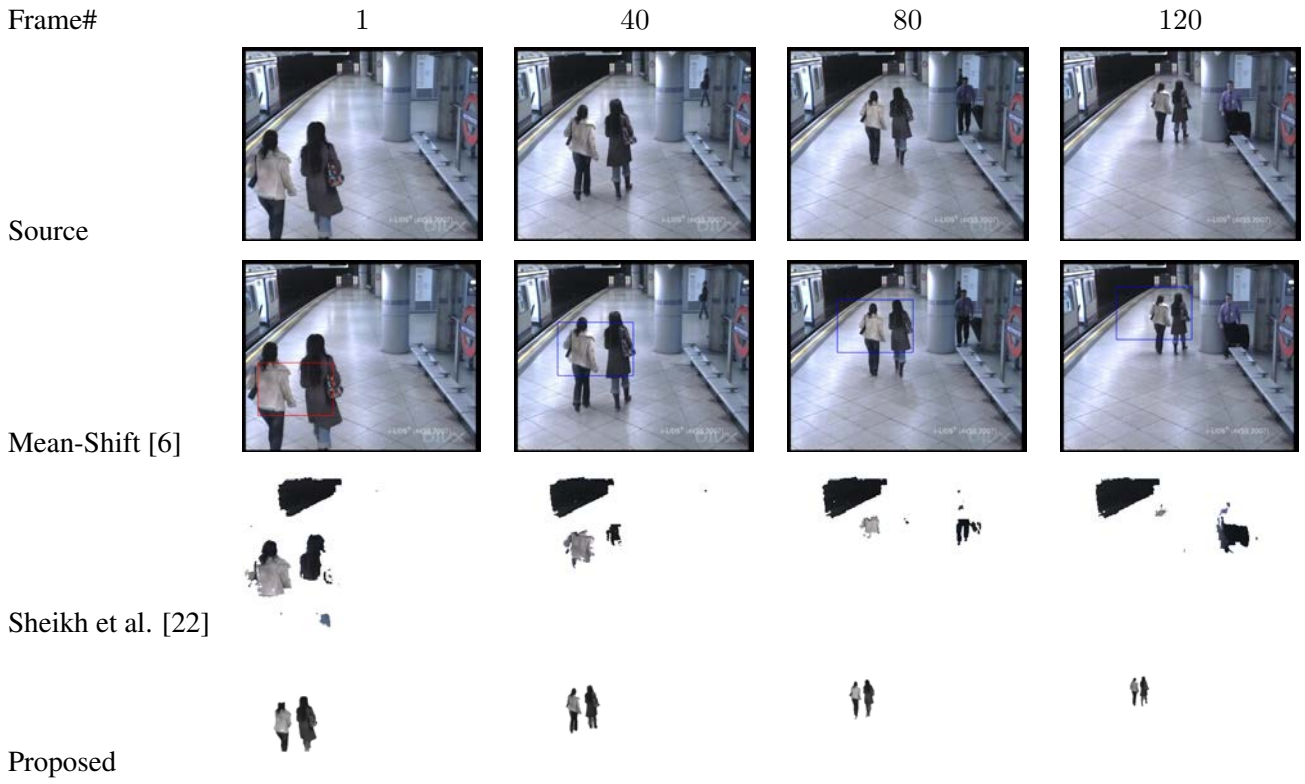


Figure 8: Tracking multiple subjects having similar colors to the background. The color of the woman's coat is similar to the background's color. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

gait analysis. Moreover, the contour extraction improved the scheme robustness by better estimating the object's color model, by avoiding the inclusion of background pixels. Compared to Sheikh and Shah's approach, the proposed scheme proved superior in both qualitative and quantitative test. In fairness, their approach provides both detection and tracking, while our requires initialization.

7 Conclusions

In this work we presented a novel probabilistic framework for integrating multiple tracking cues. Our approach utilized five different tracking cues by encoding them in an MRF, and applying a probabilistic inference scheme based on spectral relaxation, to approximate the inference that is NP-hard. The video frames are represented by superpixels, allowing to derive robust statistical representations based on Gaussian and GMM models. Thus, we derive unary and pairwise tracking cues that are robust to dynamic

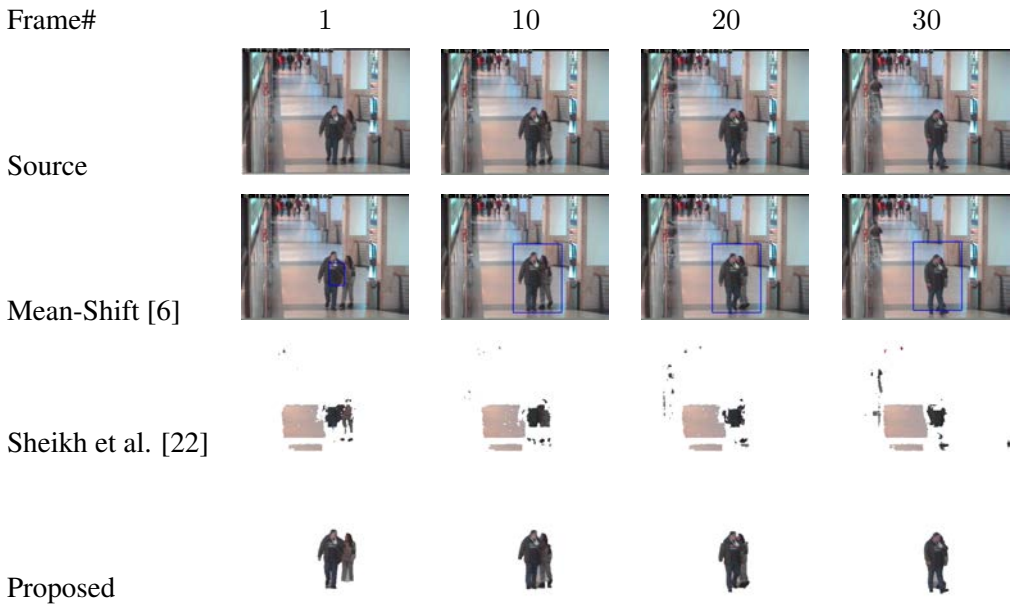


Figure 9: Tracking two objects simultaneously. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

lighting and appearance changes. The proposed scheme is experimentally shown to compare favorably with contemporary schemes such as Mean-Shift [6] and Sheikh and Shah's [22] approach. It can be extended by introducing additional tracking cues, such as change detection [24] and Kalman filtering.

References

[1] S. Alpert, M. Galun, A. Brandt, and R. Basri. Image segmentation by probabilistic bottom-up aggregation and cue integration. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(2):315 –327, feb. 2012.

[2] S. Avidan. Ensemble tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(2):261 –271, feb. 2007.

[3] B. Babenko, M.-H. Yang, and S. Belongie. Robust object tracking with online multiple instance learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1619–1632, Aug. 2011.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

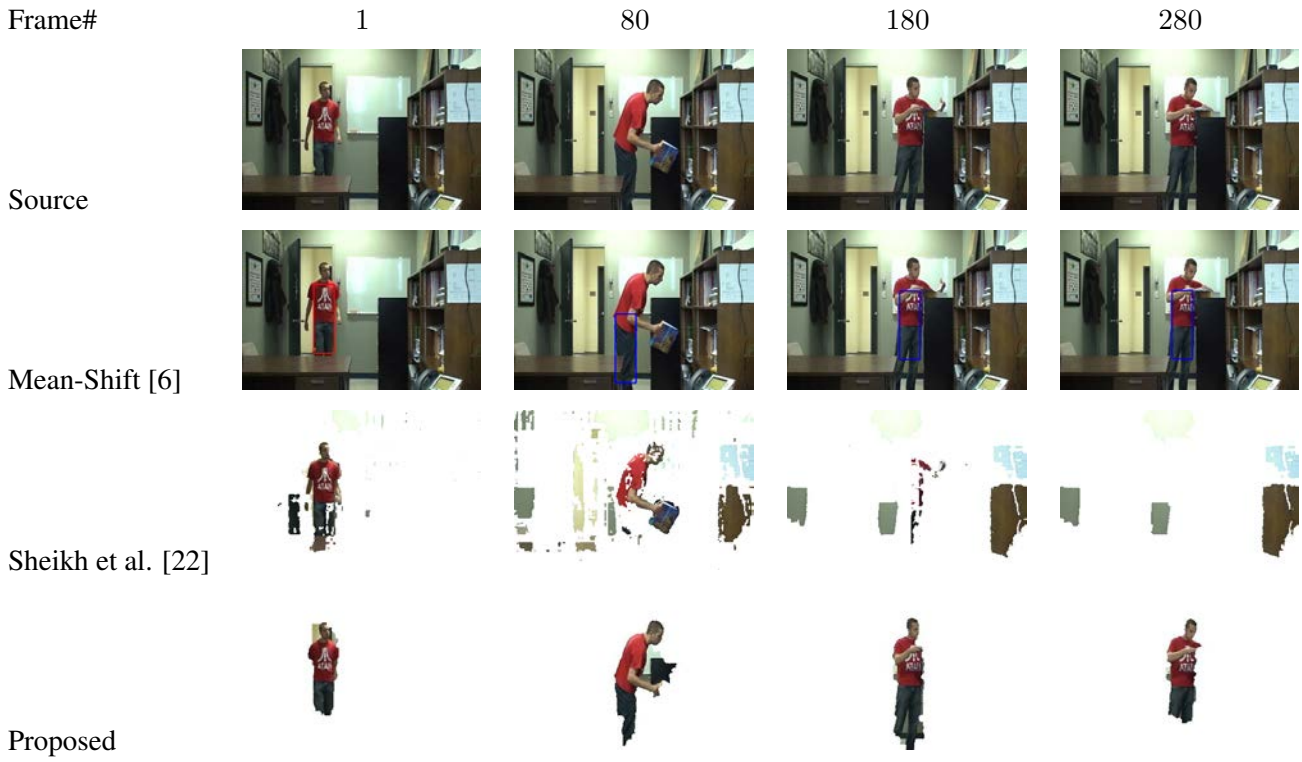


Figure 10: The tracking of an object undergoing a sudden movement. The second shows the Mean-Shift [6] results, that are depicted by a blue bounding box.

[5] R. Collins, Y. Liu, and M. Leordeanu. Online selection of discriminative tracking features. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10):1631 –1643, oct. 2005.

[6] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(5):564 – 577, may 2003.

[7] D. Corrigan, S. Robinson, and A. Kokaram. Video matting using motion extended grabcut. In *Visual Media Production, 5th European Conference on*, pages 1 –9, nov. 2008.

[8] A. Criminisi, G. Cross, A. Blake, and V. Kolmogorov. Bilayer segmentation of live video. In *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, volume 1, pages 53 – 60, june 2006.

[9] A. Egozi, Y. Keller, and H. Guterman. A probabilistic approach to spectral graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):18 –27, jan. 2013.

[10] J. Goldberger, S. Gordon, and H. Greenspan. An efficient image similarity measure based on

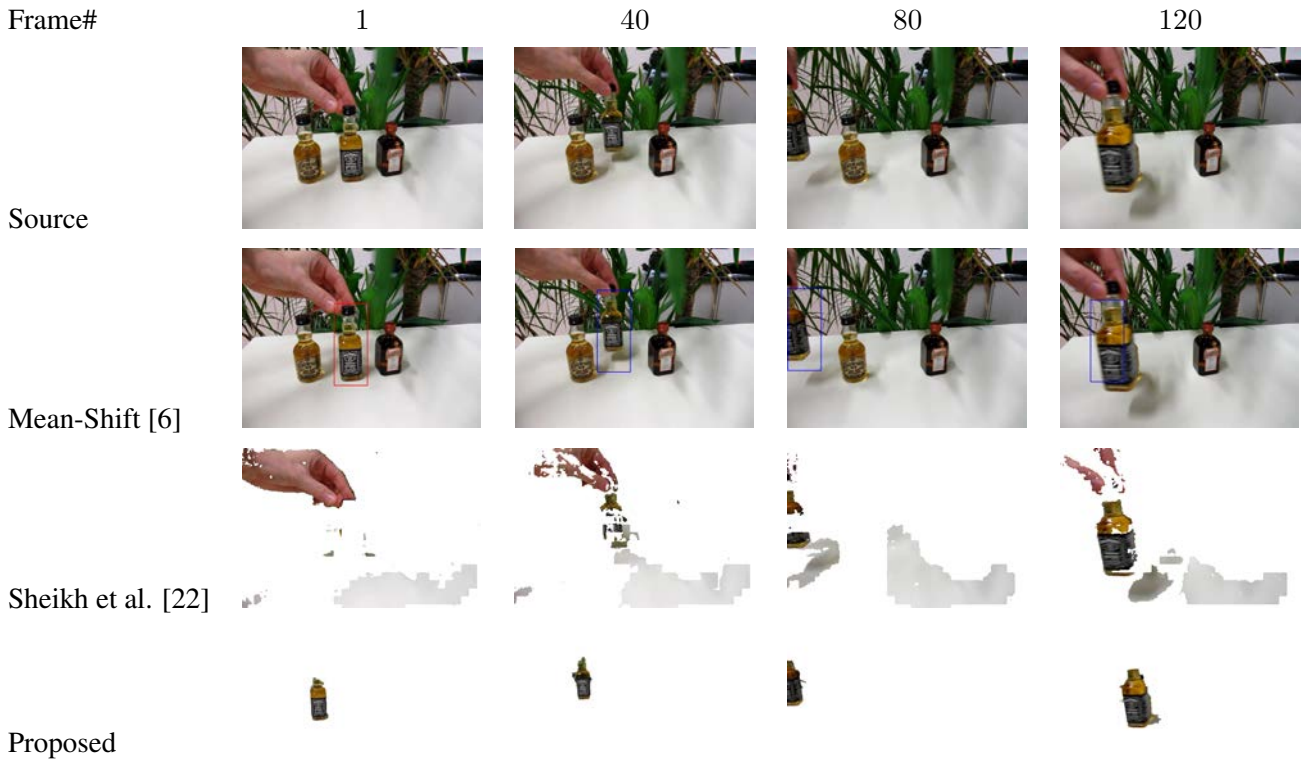


Figure 11: The results of tracking an object with varying speed and appearance.

approximations of KL-divergence between two gaussian mixtures. In *Computer Vision, Ninth IEEE International Conference on*, pages 487 –493 vol.1, oct. 2003.

[11] G. H. Golub and C. F. V. Loan. *Matrix Computations*. The Johns Hopkins University Press, 3rd edition, 1996.

[12] N. Goyette, P. Jodoin, F. Porikli, J. Konrad, and P. Ishwar. changedetection.net: A new change detection benchmark dataset. In *Change Detection (CDW-12), IEEE Workshop*, Jun. 2012.

[13] A. Jepson, D. Fleet, and T. El-Maraghi. Robust online appearance models for visual tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(10):1296 – 1311, oct. 2003.

[14] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, Mar. 1951.

[15] B. Leibe, K. Schindler, and L. Van Gool. Coupled detection and trajectory estimation for multi-object tracking. In *Computer Vision, IEEE 11th International Conference on*, pages 1 –8, oct. 2007.

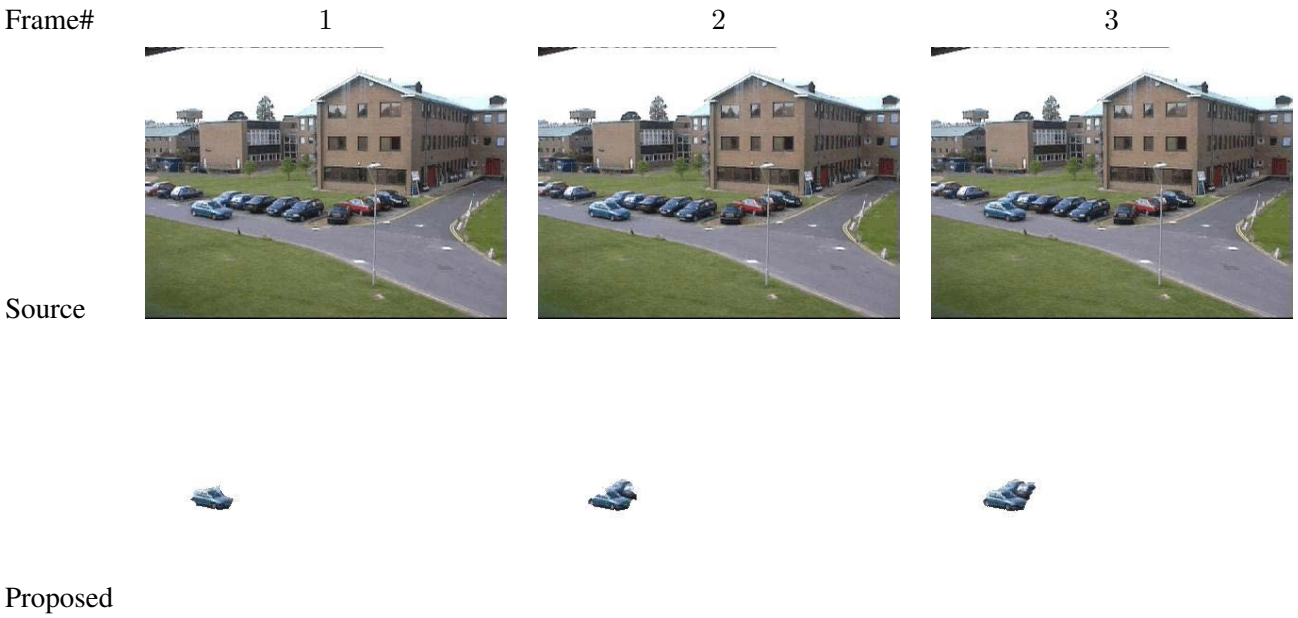
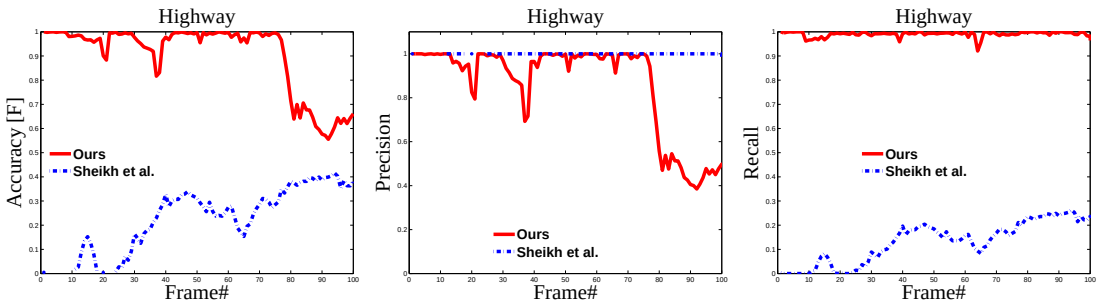


Figure 12: A tracking failure example. Due to the similar colors of the tracked and parking cars, the proposed scheme identifies both in second and third frames as the tracked object.

[16] C. Leistner, H. Grabner, and H. Bischof. Semi-supervised boosting using visual similarity learning. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1 –8, june 2008.

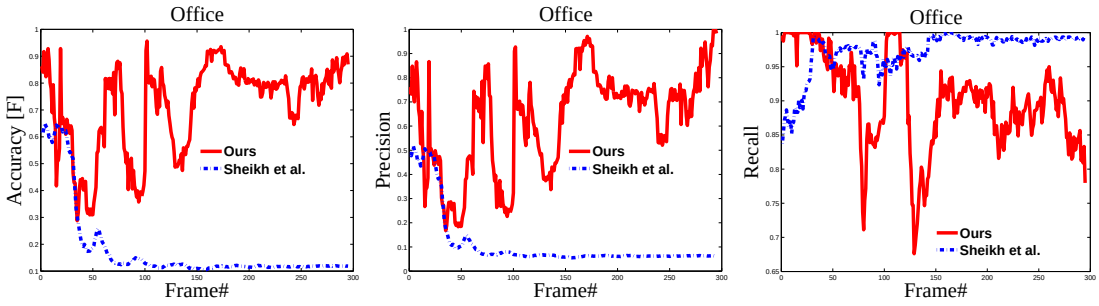
[17] M. Leordeanu and M. Hebert. A spectral technique for correspondence problems using pairwise constraints. In *Computer Vision, Tenth IEEE International Conference on*, volume 2, pages 1482 –1489 Vol. 2, oct. 2005.

[18] A. Levinshtein, A. Stere, K. Kutulakos, D. Fleet, S. Dickinson, and K. Siddiqi. Turbopixels: Fast superpixels using geometric flows. *Pattern Analysis and Machine Intelligence, IEEE Transactions*



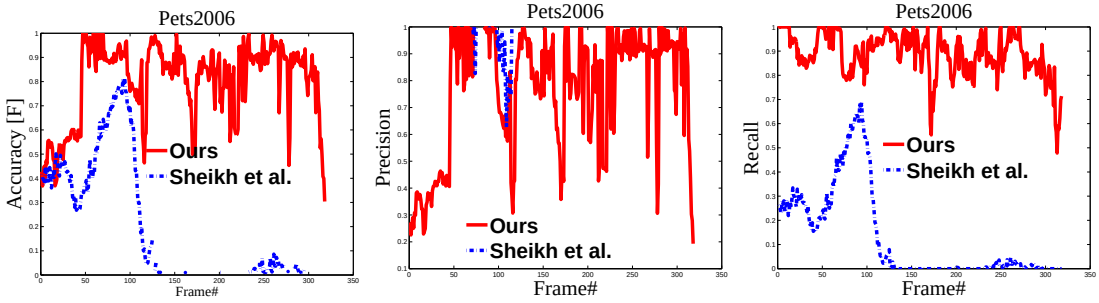
(a) Accuracy (b) Precision (c) Recall

Figure 13: Highway sequence quantitative tracking results.



(a) Accuracy (b) Precision (c) Recall

Figure 14: Office sequence quantitative tracking results.



(a) Accuracy (b) Precision (c) Recall

Figure 15: Pets2006 sequence quantitative tracking results.

on, 31(12):2290 –2297, dec. 2009.

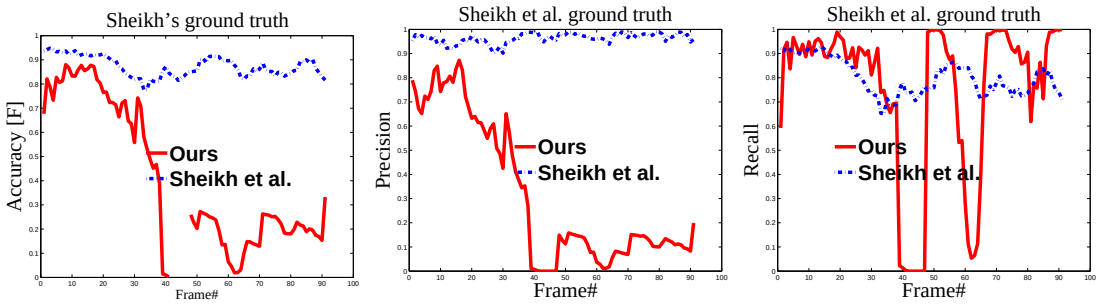
[19] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade. Tracking in low frame rate video: A cascade particle filter with discriminative observers of different life spans. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(10):1728 –1740, oct. 2008.

[20] X. Ren and J. Malik. Learning a classification model for segmentation. In *Computer Vision, Ninth IEEE International Conference on*, pages 10 –17 vol.1, oct. 2003.

[21] X. Ren and J. Malik. Tracking as repeated figure/ground segmentation. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 1 –8, june 2007.

[22] Y. Sheikh and M. Shah. Bayesian modeling of dynamic scenes for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(11):1778 –1792, nov. 2005.

[23] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2 edition, 2003.



(a) Accuracy (b) Precision (c) Recall

Figure 16: Quantitative tracking results of Sheikh and Shah's [22] groundtruth sequence.

[24] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 2 vol. (xxiii+637+663), 1999.

[25] D. Tsai, M. Flagg, A. Nakazawa, and J. M. Rehg. Motion coherent tracking using multi-label mrf optimization. *International Journal of Computer Vision*, pages 1–13, 2011.

[26] A. Vedaldi and S. Soatto. Quick shift and kernel methods for mode seeking. In *Computer Vision, 10th European Conference on*, pages 705–718, 2008.

[27] L. Vincent and P. Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(6):583 –598, jun 1991.

[28] H. Wang, D. Suter, K. Schindler, and C. Shen. Adaptive object tracking based on an effective appearance filter. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(9):1661 –1667, sept. 2007.

[29] Y. Zha, Y. Yang, and D. Bi. Graph-based transductive learning for robust visual tracking. *Pattern Recognition*, 43:187–196, January 2010.