Tracking and Recognition: A Unified Approach on Tracking and Recognition

Ms. Anuja V. Vaidya
Dept. of Electronics & Communication
Dr. J.J. Magdum College of Engg. Jaysingpur,
Maharashtra, India

Dr. Mrs. S.B. Patil
Dept of Electronics & Communication
Dr. J.J. Magdum College of Engg. Jaysingpur,
Maharashtra, India

Abstract— This paper proposes a unified approach on tracking and recognition .Object tracking is done at low level and recognition is done at high level. Traditional tracking methods give importance to low level image correspondences between frames. High level image correspondences are used for reliable tracking. Online and Offline models are used for both tracking and recognition which is done simultaneously. Thus high level offline model is combined with low level online model to increase the tracking performance. Onine model used for tracking is given to the video based recognition and at same time offline model plays important role to recognize the category of the object. This method is useful to handle difficult scenarios like abrupt change, background clutter, pose variations, occlusion and morphable objects. This is based on study of different IEEE papers.

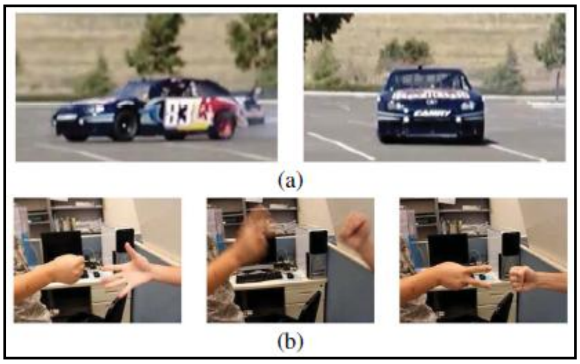
Keywords: *Object recognition, video analysis, visual tracking.*

*****

I. Introduction

Tracking is closely related to constructing correspondences between frames. Traditional tracking approaches focus on finding *low-level* correspondences based on image evidence. Online models for low-level correspondences are generally employed to adapt to the changing appearances of the target [1]–[3]. However, one notable shortcoming of these online models is that they are constructed and updated based on the previous appearance of the target without much semantic understanding. Therefore, they are limited in predicting unprecedented states of the target due to significant view changes and occlusion, and easily drift in the case when the appearance of the target changes too fast.

Figure 1 (a) shows an example where the Visual appearance of the target changes dramatically in a very short time period, making low-level image correspondences unstable.



Traditional methods may fail in these complicated scenarios, while our approach handles them well.

(a) Rapid appearance change. (b) Object morphing.
(b) Without other information, it is very likely to cause tracking failure, no matter what online model is used however, if system recognize this target as a car at a higher level, the tracking task becomes to find the same *car* in the subsequent images instead of finding the object with model is used. However, if we can recognize this target as a car at a higher level, the tracking task becomes to find the same *car* in the subsequent images instead of finding the object with same low level appearance. Therefore, the discriminative information provided by the car category, i.e., the *high-level* correspondences, can be utilized to help successfully track the target. In other words, to make tracking consistently effective in various challenging scenarios, it is necessary to combine both low-level and high-level correspondences. Some offline-trained high-level detectors with semantic meanings have already been introduced into the tracking-bydetection scheme for some specific tracking tasks, especially for human tracking [4]–[6] and vehicle tracking [7], which largely improves the tracking performance. However, these models assume the semantic meanings of targets are already known before tracking, and accordingly cannot be applied to many general applications. Consider a video surveillance scenario with a complex scene, the categories of the moving objects cannot be predicted. Nevertheless, every moving object should be correctly tracked for subsequent analysis, no matter whether it is a human, a car or even an animal. In other cases, the category of the target might change because of object morphing and camouflage (e.g., in Fig. 1(b) the states of the hand are switching between “rock”, “paper”, “scissor”), in which those pre-determined detectors are likely to fail.
(c) After all, tracking is not the final goal of video analysis but an intermediate task for some succeeding high-level processing like event detection and scene understanding. Essentially, an ideal tracking system should *actively understand* the target, and adaptively incorporate high-level semantic correspondences and low-level image correspondences. Towards this end, this paper proposes a unified approach for object tracking and recognition. In our approach, once an object is discovered and tracked,

the tracking results are continuously fed forward to the upper-level video-based recognition scheme, in which dynamic programming is adopted to recognize the category of (d) the object in the current frame. Based on the feedback from the recognition results, different off-line models dedicated to specific categories are adaptively selected, and the location of the tracked object in the next frame is determined by integrated optimization of these selected detectors and the tracking evidence. Compared with online tracking models and previous tracking-by-detection schemes, our framework has the following advantages.

1)Unlike previous tracking-by-detection methods in which the offline detectors are fixed for one category, our framework can actively recognize the target and adaptively utilize the high-level semantic information to improve the robustness of tracking. Besides, combination of object tracking and recognition is not based on the discrete, sparse output of the detectors, but achieved by an integrated optimization scheme, which accordingly makes our tracking method more flexible to difficult scenarios.

2) This approach is not only able to handle many difficult tracking scenarios such as background clutter, view changes, and severe occlusion, but also works well in some extreme situations (e.g., tracking a morphable object). Moreover, the output of our approach is further used for video understanding and event detection.

II. Related Work

Traditional online models for tracking include appearance based templates[22].(e.g., color regions[23] , and stable structures [24]), and online classifiers trained by boosting[3]. However, the efficacy of these online models heavily relies on the past tracking performance and tends to drift when the appearance of the objects keeps changing and tracking errors are accumulated. Therefore, much effort has been devoted to model updating to enhance their discriminative power and prevent the models from drifting [1], [2], [22], [25]–[29] . Nevertheless, these learning methods are still based on the immediate samples of the targets in a limited time period. If the object appearance abruptly changes to some states that have not been seen before, these models are very likely to fail. More recently, some pre-trained offline models and databases have been incorporated to the online tracking models for some specific tracking tasks. 5], [7], [30]–[32].In a human body is represented as an assembly of body parts. The responses of these part detectors are then combined as the observations for tracking. To address the occlusion problem in people tracking extract people tracklets from consecutive frames and thus build models of the individual people. When the targets are pedestrians and vehicles[7], formulates object detection and space-time trajectory estimation in a coupled optimization problem. However, these methods assume the categories of the targets are known before tracking, which is quite a strong assumption. When the objects of interests are unknown, such prior knowledge would not be available. There are also some work aiming to perform simultaneous tracking and recognition [33]–[37].Particle filter embeds the motion model and appearance model in a particle filter for face tracking, and constructs the intra- and extra-personal spaces for recognition. SURFTrac [36] tracks the objects by interest point matching and updating, and then continuously extracts feature descriptors for recognition. [37] Rotation-Invariant Fast Features (RIFF) are used for unified tracking and recognition. [35]MCMC particle filter is exploited for long-term outdoor multiobject simultaneous tracking and classification. However, these methods still treat tracking and recognition as independent steps and use conventional tracking approaches without the help of the higher-level recognition feedback. Different from this scheme, our method focuses on the information fusion of recognition and tracking, in which the recognition results are fed back to select different models and combine them in a unified optimization framework; and the tracking results are meanwhile fed forward to the recognition system, which hereby forms a closed-loop adaptation. Moreover, the recognition modules in these methods are different from the standard approaches in object recognition literature, because these methods recognize specific object instances (e.g., people identification) rather than object categories. On the contrary, we focus on semantic object recognition which yields object categories as the output.

III. FORMULATION

A. *Overview Description*

The framework of the proposed method is shown in Fig. 2.The object of interest is initialized by a user-specified bounding box, but its category is not provided. This target may or may not have a semantic meaning. Therefore, in the first few frames when the tracker does not know the target category, tracking the target only relies on the online target model, which is the same as traditional tracking. Meanwhile, video-basedobject recognition is applied on the tracked objects.

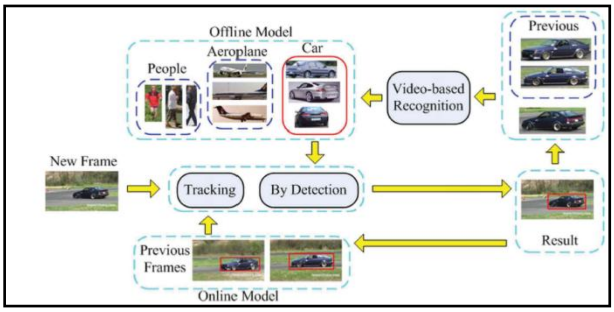


Fig. 2. Framework of the proposed method.

In the first few frames, the object category is unknown, so our tracking procedure only relies on the online target model, which is the same as traditional tracking. Meanwhile, video-based object recognition is applied on the tracked objects. When the target is recognized properly, the offline target model will be automatically incorporated to provide more information about the target.

When the target is recognized properly, the offline target model will be automatically incorporated to provide more information about the target. At time t, we denote the target state by $\mathbf{x}_t$, and the target category by ct .1 Denote the image sequence by $I_t = \{\mathbf{I}_1, \ldots, \mathbf{I}_t\}$, where $\mathbf{I}_t$ is the input image at time t. So the target measurement at time t is $\mathbf{z}_t = \mathbf{I}_t(\mathbf{x}_t)$. In a visual tracking framework, the online target model is generally based on low-level features. We denote the online target model by M^{L}_{t}= _($\mathbf{z}_1, \ldots, \mathbf{z}_t$), where _ is a mapping (e.g., extracting feature descriptors from the target). For the object category, we consider there are N different object classes (denoted by $C^1, \ldots ,C^N$). For the regions not belonging to any known class or even with no semantic meanings, we denote by C^0 the complementary set, namely the "others" class. Hence, $c_t \in \{C^0, C^1, \ldots, C^N\}$. Each object class C^{i} is associated with a specific offline model ($M^H{}_C i$), which is an abstraction of a specific object class[2].At time t, our objective is to estimate $\mathbf{x}_t$ and c_t , based on the input image sequence I_t as well as the offline model.

Generally, it is very difficult to estimate $\mathbf{x}_t$ and c_t simultaneously. Therefore we employ a two-step EM-like method here: at time t, we first estimate $\mathbf{x}_t$ (i.e., "tracking"), and then estimate ct based on the new tracking result $\mathbf{z}t = \mathbf{I}_t(\mathbf{x}_t)$ (i.e., "recognition"). In the next subsections, we will present these two steps in details.

B. Tracking Procedure

Different from traditional tracking, the estimation of $\mathbf{x}_t$ in our approach is based on the online target model $M^L\ t-1$, the offline model $M^H{}_{ct\text{-}1}$ selected by the previous recognition result c_{t-1}, and the current input image $\mathbf{I}_t$. In a Bayesian perspective, we have

$$\mathbf{x}^*_{\mathbf{t}} = \arg\max_{x_{t \in \Omega}} p(\mathbf{x}_{\mathbf{t}\,|}\, \mathbf{M}_{\mathbf{ct}}\text{-}1^{\mathrm{H}}, \mathbf{I}_{\mathrm{t}})$$

$$= \arg\max_{x_{t \in \Omega}} p(\mathbf{M}^{\mathbf{L}}{}_{\mathbf{t}}\text{-}1, \mathbf{M}_{\mathbf{ct}}\text{-}1^{\mathrm{H}} |\mathbf{x}_{\mathrm{t}}, \mathbf{I}_{\mathrm{t}})p(\mathbf{x}_{\mathbf{t}\,|}\, \mathbf{I}_{\mathrm{t}})$$

$$= \arg\max p(\mathbf{M}^{\mathbf{L}}{}_{\mathbf{t}}\text{-}1|\, \mathbf{x}_{\mathbf{t}}, \mathbf{I}_{\mathrm{t}})\, p(\mathbf{M}_{\mathbf{ct}}\text{-}1^{\mathrm{H}} |\mathbf{x}_{\mathrm{t}}, \mathbf{I}_{\mathrm{t}})p(\mathbf{x}_{\mathbf{t}\,|}\, \mathbf{I}_{\mathrm{t}})$$

$$\mathbf{X}^*_{\mathbf{t}} = \arg\max p(\mathbf{M}^{\mathbf{L}}{}_{\mathbf{t}}\text{-}1|\, \mathbf{x}_{\mathbf{t}}, \mathbf{I}_{\mathrm{t}})\, p(\mathbf{M}_{\mathbf{ct}}\text{-}1^{\mathrm{H}} |\mathbf{x}_{\mathrm{t}}, \quad \mathbf{I}_{\mathrm{t}}). \qquad –(1)$$

In the third equation of Eq. 1,

we assume $M^L\ t-1$ and $M^H\ ct-1$

are conditionally independent given image $\mathbf{I}_t$ and position $\mathbf{x}_t$, because $M^L{}_{t-1}$ can be viewed as the target online appearance variation, and $M^H\ ct-1$ is related to the intrinsic appearance of the target. This argues that the two models are independent given the image observations. C_{t} does not depend on the online tracking model, once the image measurement $\mathbf{I}_t(\mathbf{x}_t)$ is given. The last equation means that we consider $p(\mathbf{x}_t \mid \mathbf{I}_t) = 1/|\mathbf{\Omega}|$ is a uniform distribution in the search space of the target. When the recognition result is not available

($c_{t-1} = C^0$), the problem is simplified as

$$\mathbf{x}^*\ t = \underset{\mathbf{x}_t \in \Omega}{\mathrm{argmax}}\ p(\mathbf{x}t\, |M^L{}_{t-1}, \mathbf{I}_t),$$

where only the online object model is considered. For every frame, the online target model is updated as $M^L{}_t = \Psi(\mathbf{z}_1, \ldots, \mathbf{z}_t)$.

The state $\mathbf{x}_t = \{x, y, w, h\}$ consists of the target central position (x, y), its width w, and its height h. Maximizing the likelihood term in Eq. 1 can be formulated as an energy minimization problem by defining the energy term $E = -\ln p$. Therefore,

$$\mathbf{X}^*_{\mathbf{t}} = \arg\min_{X_{t \in} \Omega} E(\mathbf{x}t) = \arg\min Et(\mathbf{X}_t) + Ed(\mathbf{X}_t) \qquad \text{-(2)}$$

where $Et(\mathbf{x}t) = -\ln p(M^{\mathrm{L}}\ t-1|\, \mathbf{X}_t, \mathbf{I}_t)$ is the energy term related to tracking, and

$Ed(\mathbf{x}_t) = -\ln p(M^H{}_{ct-1} |\mathbf{x}_t, \mathbf{I}_t)$ is the energy term related to detection. The term "detection" is consistent with the widely used term "tracking-by-detection" in state-of the- art literature. Please note that we absorb the normalization factors into the energy terms without confusion. Both energy terms are further decomposed.

1) *Tracking Term:*

The widely used correspondences in object tracking are point correspondences and region correspondences.The point correspondences reflect the local Information, while the region correspondences reflect the global information. As contour correspondences may not always be reliable in cluttered situation, we do not employ them. here. Therefore, the tracking term $E_t(\mathbf{x}_t)$ can be written as the weighted sum of the energy terms of these two types.

i.e., $E_t(\mathbf{x}_t) = w_s E_s(\mathbf{x}_t) + w_h E_h(\mathbf{x}_t)$.

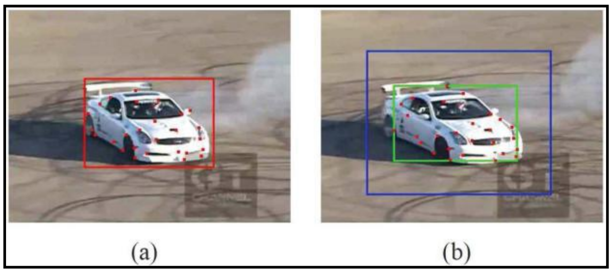


Fig. 3. Determine the search range. (a) Previous frame. The red bounding box shows the target location at the previous frame. (b) Current frame.The red points are corresponding matching points between two frames. The green box is the inner boundary, while the blue one is the outer boundary.

For point correspondences, we choose Harris-corner points with SIFT descriptor as our salient points, and denote the set of the salient points on the target at time *t* by s^t_i . For each salient point s^{i}_{t-1} on the target, we record its relative position w.r.t the target center as $\mathbf{l}^{i}_{t-1}$, normalized by the target size. And for each s^{i}_{t-1} , we find its correspondence b^{t}_{i} at time *t* by SIFT matching, with the matching error w^{t}_{i} (Fig. 3 givesone example of corresponding matching points, shown by red points). Given the candidate region $\mathbf{x}_t$, the relative position of b^{t}_{i} is uniquely determined, denoted by $\mathbf{g}^{t}_{i}$. We assume that the relative position of the same salient point w.r.t the target cannot change rapidly. Ideally, if the target movement is only translation or scaling, $E_s = 0$. So *Es* is related to the deformation of the target. For region correspondences, we employ the color histogram matching. We obtain the target histogram h^{t-1} at time $t - 1$ in the HSV color space.

Where ||·|| is the L_2 norm (outlier saturated). The target histogram $h(\mathbf{x}_t)t$ can be quickly computed with an integral image.

2) Detection Term:

$Ed\ (\mathbf{x}_t)$ measures the difference between the target candidate $\mathbf{x}_t$ and the specific offline model $M^H{}_{\text{ct-1}}$ of class c_{t-1}. This is quite related to image object recognition. We define a cost $U(d, c) = -\ln p(d \mid c)$ where *d* is the measurement of the target instance and $c \in \{C^0, C^1, \ldots, C^N\}$ is a specific class. The object detection for a specific class *c* is indeed

$\mathbf{x}^* = \text{argmin } U(d(\mathbf{x}), c)$, while the recognition

$\mathbf{x} \in \Omega$

procedure is formulated as finding the best

c^*such that $c^* = \arg \quad \min \quad U(d, c).$

$c \in \{C0, C1, \ldots, CN\}$

Therefore, we use the same cost function . $U(d, c)$ for object detection and recognition.In object detection, an exemplar-based energy term $Ed\ (\mathbf{x}_t) = U(\mathbf{z}_t, c_{t-1})$ (recall $\mathbf{z}_t = \mathbf{I}_t\ (\mathbf{x}_t)$) is designed for each specific class c_{t-1}, which can be decomposed as the weighted.

sum $Ed\ (\mathbf{x}t) = w_p\ E_p(\mathbf{x}t) + w_e E_e(\mathbf{x}_t)$, where $Ep(\mathbf{x}_t)$ is related to the pyramid matching of salient points , and $E_e(\mathbf{x}_t)$mis related to the pyramid matching of the histograms of edge directions:

$$E_p(\mathbf{x}_t) = 1/r \sum_{j=1}^{r} \| \mathbf{f}_p(\mathbf{z}_t) - N\,N\,j,\mathrm{C}_{t-1}\, \mathbf{f}_p(\mathbf{z}_t) \|2 \qquad \text{--- (5)}$$

$$E_e(\mathbf{x}_t) = 1/r \sum_{j=1} \| \mathbf{f}_e(\mathbf{z}_t) - N\,N\,j,\mathrm{C}_{t-1}\, \mathbf{f}_e(\mathbf{z}_t) \|2 \qquad \text{---(6)}$$

where $\mathbf{f}_p(\cdot)$ and $\mathbf{f}_e(\cdot)$ are the features extracted for spatial pyramid matching of SIFT descriptors and edge histograms, respectively (more details are given in Sec. III-C). $NNj,ct{-}1$ $\mathbf{f}_p(\mathbf{z}_t)$ is the *j* th nearest neighbor of $\mathbf{z}_t$ in the feature space $\mathbf{f}_p$ from the training samples of class c_{t-1}. Eq. 6 is defined in the same way. Please note that the edge histograms for all hypothesis can be quickly computed with the integral image. Both terms are commonly used in object detection methods [39]. When $c_{t-1} = C0$, the offline model is not activated and thus $\mathrm{E_d}(\mathbf{x}_t)=0$.

3) Optimization Method: As the optimization problem $\mathbf{x}^*{}_t = \arg\min E(\mathbf{x}t)$

$\mathbf{X}t \in \Omega$

does not have an analytic solution, we obtain $\mathbf{x}^*{}_t$ via exhaustive search in $\Omega^{4.}$ To reduce the search range, we perform a coarse-to-fine search in the space _. We construct a subset $\Omega^{'} \subset \Omega$ with spacing *m* pixels, and define

$\mathbf{x}^{**}{}_t = \arg\min \ Et\ (\mathbf{x}t) + Ed\ (\mathbf{x}_t)$

$\mathbf{x}t \in \Omega$

which is a suboptimal solution. Then start from

$\mathbf{x}^{**}{}_t$ and perform the local search every $m / 2$ pixels, and the local optimum is treated as our tracking result. The parameter *m* depends on the target size, and we set $m = 10$ for the general case. The search range Ω is determined as follows. At time *t*, have obtained the matching points $\mathrm{b}^{\mathrm{t}}{}_{\mathrm{i}}$ in the target. Therefore, for any candidate region $\mathbf{x}_t$, $b^{t}{}_{I} \in \mathbf{x}t$. This gives the inner boundary of the candidate regions. For the outer boundary, Estimate the rough global motion $(\Delta x_t, \Delta y_t)$ of the target by simply averaging the motion of each salient point. The outer boundary is then the rectangle with the center $(xt-1+\_xt\ ,\ yt-1+\_yt\ )$ and the width/height $wt-1+(wt-1+\ ht-1)/4,\ ht-1\ +(wt-1\ +ht-1)/4$. The illustration is shown . In Fig. 3, the candidate bounding box should contain the green box, and should be contained in the blue box..

C. Video-Based Object Recognition

Given the current target state $\mathbf{x}_t$, compute the target category c_t based on the target measurement $\mathbf{z}_t = \mathbf{I}_t\ (\mathbf{x}_t\ )$. This can be formulated as an object recognition problem in a single image. However, recognition in a single image may not achieve good performance. As the decision is only made on one view of the target, recognition can be difficult due to complex situations such as partial occlusion. Therefore, we instead find the optimal sequence $\{c_1, . . . , c_t\}$ given the measurement

$\mathbf{z}_1, . . . , \mathbf{z}_t$, which is indeed the *video-based object recognition*.

$$C_t = \{C_1,\ldots\ldots.C_t\}$$

$$Z_t = \{Z_1,\ldots\ldots..Z_t\}$$

$$\{C_1^*,\ldots\ldots.., C_t^*\} = \arg\max p(C_t \mid Z_t)$$

$$= \arg\max p(Z_t \mid C_t)\ p(C_t)$$

$$= \arg\max \mathop{\boldsymbol{\pi}}_{\mathbf{i-1}}^{\mathbf{t}} p(z_i \mid C_i)\ p(C_i \mid C_{i-1}) \qquad \text{---(7)}$$

The last equation assumes that $\mathbf{Z_i}$ are independent of $\mathbf{Z}_j$ and C_j (i not equal to j) is a Markov chain. Finding the optimal sequence $\{ci\}$ in Eq. 7 is equivalent to the problem of finding the best hidden state sequence in an HMM. We can employ the Viterbi algorithm (indeed a dynamic programming approach) for the inference. In practice, at time t, we do not have to estimate the sequence $\{c1, . . . , ct\}$ starting from the first frame because of the computational complexity. Instead, we construct a time window which only considers the recent T frames, i.e., the sequence

$\{c_{t\ -T}+1, . . . , ct\ \}$. The prior term is

$p(c_{\ t-T+1}) = p(\ c\ t-T+1|c_{\ t-T})$

where $c_{\ t-T}$ is known.

As above, we model the probability $p(c_i\ |c_{\ i-}1)$ and $p(\mathbf{z}_i\ |c_i\ )$ by using the energy terms: the transition cost $V\ (c_i\ ,\ c_{\ j}\ ) = -\ln\ p(c_i\ \ |c_{\ \ j}\ \ )$ which measures how likely the state $c\ j$ switches to c_i (it can be manually set as fixed values), and the cost

$E_r\ (\mathbf{z}_i\ ,\ c_i\ ) = -\ln\ p(\mathbf{z}_i\ |c_i\ )\ \_\ U(\mathbf{z}_i\ ,\ c_i\ )$, which is related to object recognition in a single frame.We use the same $U(\cdot, \cdot)$ as described in Sec. III-B, This is the Naive-Bayes NN classifier, which overcomes the inferior performance of conventional NN-based image classifiers. Now we give some more details in obtaining $\mathbf{f}p(\cdot)$ and $\mathbf{f}e(\cdot)$.In the salient point representation, we extract Harris-corner points with SIFT descriptors, and quantize them using a 300 entry codebook that was created by K-means clustering a random subset of 20,000 descriptors. We use a two-level pyramid to construct the feature $\mathbf{f}p(\cdot)$.To construct a histogram of edge directions, we use [−1, 0, 1] gradient filter with no smoothing, and nine different directions are extracted. For color images, we compute separate gradients for each color channel,and take the one with the largest norm as the pixel's gradient vector .The edge histogram $\mathbf{f}e(\cdot)$ is represented using a uniformly weighted spatial pyramid with three levels [38]. KD-tree is used for the efficiency of NN search in order to reduce the computational complexity . We choose $r = 15$ in our experiment. The training images for object recognition are collected from PASCAL VOC Challenge 2007 data set [43]. We consider some often seen moving objects as object classes. Specifically, we consider six classes: aeroplane (A), boat (B), car (C), people (P), quadruped (Q), and others (O, i.e., C^0). The "quadruped" class includes horse/cat/dog, because the shape of these animals is very similar in many cases. For the "people" class, as the Pascal VOC 2007 data[43] set includes various people postures like sitting, which is not good for recognition of moving persons, we use the training samples from INRIA dataset instead. The class $C0$ includes some static object classes: chair/sofa/table/monitor. We also include some natural scene images into this class. The natural scene images are from [44]. In order to avoid wrong recognition result, the object is recognized as class C^0 if $p(\mathbf{z}_i\ |c_i\ )$ is low for all C_i , $i = 1, . . . , N$. Then our tracking procedure is simplified as

$$\mathbf{x}^*_t = \mathop{\text{argmax}}_{\mathbf{x}_t \in \Omega}\ \ p(\mathbf{x}_t\ |M^L_{\ t-1}, \mathbf{I}_t\ ).$$

1) ***Tracking Morphable Objects****:* The target category c can be extended to describe different status of the object. For example, if to track morphable objects, we regard $\{C^1, . . . ,C^N\}$ as the different status of the object, and C^0 as the "others" status. Then the parameter ct describes the object status at time t. By adjusting the

transition cost *V* (c_i , c_j), the proposed recognition scheme can be easily applied to this scenario.

IV. EXPERIMENTS

In this section, we first go through the technical details. Then we compare the proposed video-based object recognition method with still-image object recognition, followed by the sensitivity analysis of the parameters. The tracking performance in various scenarios is then evaluated.

A. Technical Details

For technical details, the parameters w_s , w_h, w_p, and w_e are chosen to make the energy terms comparable. In practice, the weight of each term is adaptively adjusted by a confidence score, which measures the performance of one term at current frame. Likewise define the confidence score for other energy terms. The thresholds for the offline model (e.g., T_p and T_e) are determined based on the training data. Different categories have different thresholds. For a certain category ˆ *c*, we compute

T_p(ˆ *c)* as follows (similarly for T_e(ˆ *c*)):

Select one image from the training data, and collect some image regions close to the target location as the positive samples. For each image region **z**, we obtain the average distance d_p = E_p(**z**, ˆ *c)* between this region and its nearest neighbors from remaining training samples. Intuitively, d_p is less than T_p(ˆ *c*), since it is the positive data. For all the training images, and all the positive image regions we collected for each image, then compute d_p similarly. Therefore, obtain the distribution of d_p. Similarly, we collect negative samples which are far from the labeled image region in each image, and obtain the distribution of *dn* for negative samples. So T_p(ˆ *c)* is essentially the Bayes optimal decision boundary, and it can be easily obtained numerically based on these two distributions. The thresholds for the online model (e.g., T_s and T_h) are determined empirically.

Although use same energy terms for all classes, our algorithm is flexible in that the energy term $E_d(\mathbf{x}_t)$ can have different choices for different object classes. We find that “people” is a special object class, because the SVM classifier for object detection usually works well in this class. However it is not always the case for other classes. Therefore, in case of human detection, we change the detection term to the energy term of a linear SVM classifier. We also choose the HOG feature for “people” class. The object detection is essentially a “one-against-others” classifier, where this task is simpler than the object recognition task. So we can simplify the detection term by discarding the energy term E_p from E_d, so as to reduce the computational complexity. In object recognition, E_p is still included. As the pyramid matching of the salient points only needs to compute once (at the tracked bounding box), the complexity is mild.

B. Recognition Evaluation

Consistent tracking improves the recognition performance in our approach.. The reason is that we are dealing with video-based recognition task, rather than an image-set-based one. Hence, the transition and continuity in the video frames are important clues, which are properly adopted in our method.

C. Sensitivity Analysis

The values of the normalizing constants (*ws* , *wh*, *wp*, and *we*) are determined empirically. We tried different thresholds (±20%) in the experiments, and most results vary in a small range . The reason is that the object recognition on our selected object classes is very successful in literature, and our online SIFT/histogram matching has superior performance in accurately localizing the target.

D. Tracking Evaluation

Many difficult real videos containing various object classes. Most videos are downloaded from Youtube. The algorithm is implemented in Matlab and runs 2 ~ 0.5 frames per second on average depending on the object size. We compared the performance using online model alone, offline model alone, and both combined in our algorithm. The combination of online and offline models performs better than either model alone. The online model does not handle large view changes, while the offline model does not always achieve the correct localization. However, when combined, these two models compensate for each other. We also compared our method with five state-of-art online learning trackers, i.e., the multiple instance learning (MIL) tracker , the online Ada Boost (OAB) tracker, visual tracking decomposition (VTD) tracker discriminative intentional tracker (DAT).Metric differential tracker (MDT). Since the object class information is *unknown* in the beginning, which makes the offline learning based methods (like people tracking-by-detection) infeasible. For the recognition part is merely object identification which simply matches the target to one image in the database, which is completely different from our method. Therefore, we did not include those methods for comparison. In addition to obtaining the target location, our method also recognizes the target at every frame. The recognition feedback introduced. Note that we can deal with aspect change, as our parameter space is *(x, y, w, h).*

The baseline online trackers can only track the local region of the white dog, as the low-level feature information is not enough to estimate the correct scale in this case. In contrast, high-level semantic informationto generic object tracking. Through the above discussions,we summarize the limitations in our method: (1) The ambiguity of the tracking problem increases, as the number of object categories increases. (2) The wrong recognition result probably leads to error propagation. (3) The current design may not be appropriate for some tracking dataset, due to data type inconsistency limitations may be resolved via the progress on robust detection, or the progress on large scale robust object recognition.

V. CONCLUSION

As a mid-level task, visual tracking plays an important role for high-level semantic understanding or video analysis. Meanwhile the high-level understanding (e.g., object recognition) should feed back some guidance for low-level tracking. Motivated by this propose a unified approach to object tracking and recognition. In framework, once the objects are discovered and tracked, the tracking result is fed forward to the object recognition module. The recognition result is fed back to activate the off-line model to and help improve tracking. Extensive experiments demonstrate the efficiency of the proposed method.

REFERENCES

[1] S. Avidan, “Ensemble tracking,” *IEEE Trans. Pattern Anal. Mach. Intell.*,vol. 29, no. 2, pp. 261–271, Feb. 2007.

[2] B. Babenko, M.-H. Yang, and S. Belongie, “Visual tracking with online multiple instance learning,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 983–990.

[3] H. Grabner, M. Grabner, and H. Bischof, “Real-time tracking via on-line boosting,” in *Proc. British Mach. Vis. Conf.*, 2006, pp. 1–10.

[4] 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,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1728–1740, Oct. 2008.

[5] B. Wu and R. Nevatia, “Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors, *Int. J. Comput. Vis.*, vol. 75, no. 2, pp. 247–266, Nov. 2007.

[6] C. Huang, B. Wu, and R. Nevatia, “Robust object tracking by hierarchical association of detection responses,” in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 788–801.

[7] B. Leibe, K. Schindler, N. Cornelis, and L. V. Gool, “Coupled object detection and tracking from static cameras and moving vehicles,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1683–1698, Oct. 2008.

[8] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: A survey,” *ACM Comput. Surv.*, vol. 38, no. 4, pp. pp. 1–13, 2006.

[9] M. Isard and A. Blake, “Contour tracking by stochastic propagation of conditional density,” in *Proc. Eur. Conf. Comput. Vis.*, 1996, pp. 343–356.

[10] Z. Kalal, J. Matas, and K. Mikolajczyk, “P-N learning: Bootstrapping binary classifiers by structural constraints,” in *Proc. Conf. Comput. Vis. Pattern Recognition.*, 2010, pp. 49–56.

[11] [11] A. Srivastava and E. Klassen, “Bayesian and geometric subspace tracking,”*Adv. Appl. Probab.*, vol. 36, pp. 43–56, Dec. 2004.

[12] 12] F. Porikli, O. Tuzel, and P. Meer, “Covariance tracking using model update based on lie algebra,” in *Proc. Conf. Comput. Vis. Pattern Recognition.*, 2006, pp. 728–735.

[13] A. Tyagi and J. W. Davis, “A recursive filter for linear systems on Riemannian manifolds,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[14] J. Kwon, K. M. Lee, and F. C. Park, “Visual tracking via geometric particle filtering on the affine group with optimal importance functions,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 991–998.

[15] X. Mei and H. Ling, “Robust visual tracking using _1 minimization,” in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 1436–1443.

[16] R. Li and R. Chellappa, “Aligning spatio-temporal signals on a special manifold,” in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 547–560.

[17] C. Bibby and I. Reid, “Real-time tracking of multiple occluding objects using level sets,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1307–1314.

[18] N. Alt, S. Hinterstoisser, and N. Navab, “Rapid selection of reliable templates for visual tracking,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 1355–1362.

[19] J. Fan, “Toward robust visual tracking: Creating reliable observations from videos,” Ph.D. thesis, Dept. Comput. Eng., Northwestern Univ.,Evanston, IL, 2011.

[20] X. Mei, H. Ling, and Y. Wu, “Minimum error bounded efficient _1 tracker with occlusion detection,” in *Proc. Conf. Comput. Vis. Pattern Recognition.*, 2011, pp. 1257–1264.

[21] X. Li, A. Dick, H. Wang, C. Shen, and A. Van den Hengel, “Graph mode-based contextual kernels for robust SVM tracking,” in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 1156–1163.

[22] I. Matthews, T. Ishikawa, and S. Baker, “The template update problem,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 6, pp. 810–815, Jun. 2003.

[23] G. Hager, M. Dewan, and C. Stewart, “Multiple kernel tracking with SSD,” in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2004, pp. 790– 797.

[24] A. D. Jepson, D. Fleet, and T. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1296–1311, Oct. 2003.

[25] J. Fan, X. Shen, and Y. Wu, "Scribble tracker: A matting-based approach for robust tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 8, pp. 1633–1644, Aug. 2012.

[26] J. Fan, Y. Wu, and S. Dai, "Discriminative spatial attention for robust tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 480–493.

[27] R. Li, R. Chellappa, and S. K. Zhou, "Learning multi-modal densities on discriminative temporal interaction manifold for group activity recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 2450–2457.

[28] [28] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1195–1209, Jul.2009.

[29] N. Jiang, W. Liu, and Y. Wu, "Adaptive and discriminative metric differential tracking," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1161–1168.

[30] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *Proc. Conf. Comput. Vis. Pattern Recognition.*, 2008, pp. 1–8.

[31] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in *Proc. Conf. Comput. Vis. Pattern Recognition.*, 2005, pp. 1–8.

[32] J. Gall, N. Razavi, and L. V. Gool, "On-line adaption of class-specific codebooks for instance tracking," in *Proc. 21st British Mach. Vis. Conf.*, 2010, pp. 1–12.

[33] S. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1434–1456, Nov. 2004.

[34] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman, "Visual tracking and recognition using probabilistic appearance manifolds," *Comput. Vis. Image Understand.*, vol. 99, no. 3, pp. 303–331, 2005.

[35] F. Bardet, T. Chateau, and D. Ramadasan, "Illumination aware MCMC particle filter for long-term outdoor multi-object simultaneous tracking and classification," in *Proc. Int. Conf. Comput. Vis.*, 2009, pp. 1623– 1630.

[36] D.-N. Ta, W.-C. Chen, N. Gelfand, and K. Pulli, "Surftrac: Efficient tracking and continuous object recognition using local feature descriptors," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1–8.

[37] G. Takacs, V. Chandrasekhar, S. Tsai, D. Chen, R. Grzeszczuk, and B. Girod, "Unified real-time tracking and recognition with rotationinvariant fast features," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 934–941.