Online Multi-Object Tracking via Robust Collaborative Model and Sample Selection

Mohamed A. Naiel[a], M. Omair Ahmad[a,*], M.N.S. Swamy[a], Jongwoo Lim[b], Ming-Hsuan Yang[c]

[a]*Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada H3G 1M8*
[b]*Department of Computer Science and Engineering, Hanyang University, Seoul 113-791, Republic of Korea*
[c]*School of Engineering, University of California, Merced, CA 95344 USA*

Abstract

The past decade has witnessed significant progress in object detection and tracking in videos. In this paper, we present a collaborative model between a pre-trained object detector and a number of single-object online trackers within the particle filtering framework. For each frame, we construct an association between detections and trackers, and treat each detected image region as a key sample, for online update, if it is associated to a tracker. We present a motion model that incorporates the associated detections with object dynamics. Furthermore, we propose an effective sample selection scheme to update the appearance model of each tracker. We use discriminative and generative appearance models for the likelihood function and data association, respectively. Experimental results show that the proposed scheme generally outperforms state-of-the-art methods.

Keywords: Multi-object tracking, particle filter, collaborative model, sample selection, sparse representation.

1. Introduction

Multi-object tracking (MOT) is one challenging vision problem with numerous applications in automatic visual surveillance, behavior analysis, and intelligent transportation systems, to name a few. In the past decade, more attention has been paid on detecting and tracking one or more objects in videos. Recent advancement in object detection facilitates collaboration between the detection and tracking modules for multi-object tracking [1].

Robust multi-object tracking entails solving many challenging problems such as occlusion, appearance variation, and illumination change. A pre-trained object detector robust to appearance variation of one specific class is often used as a critical module of most multi-object tracking methods. Specifically, one detector encodes the generic pattern information about a certain object class (e.g., cars, pedestrians and faces), and one tracker models the appearance of the specific target to maintain the target identity in an image sequence. However, an object detector is likely to generate false positives and negatives, thereby affecting the performance of a tracker in terms of data association and online model update.

In multi-object tracking, offline methods based on global optimization of all object trajectories usually perform better than online counterparts [2–10], and an experimental evaluation of recent methods can be found in [11]. For instance, Brendel *et al.* proposed the maximum-weight independent set of a graph for data association [3], and Zamir *et al.* used the generalized minimum clique graph to solve the data association [6]. In [10], the data association problem is solved by using a sliding window of three frames to generate short tracklets, and in case of inconsistencies, the algorithm uses larger tracklet optimization. The minimum-cost network flow is then used to optimize the overall object trajectories. For real-time applications, online methods [1, 12–14] have been developed within the tracking-by-detection framework where data association between detections and trackers are carried out in an online manner.

Table 1 summarizes the multi-object tracking methods that are most related to this work. Online multi-object tracking can be carried out by using joint state-space model for multi-targets [12, 15–19]. For instance, a mixture particle filter has been proposed [12] to compute the posterior probability via the collaboration between an object detector and the proposal distribution of the particle filter. However, the joint state-space tracking methods require high computational complexity. The probability hypothesis density filter [20] has been incorporated in visual multi-target tracking [18, 21] since the time complexity is linear with respect to the number of targets. However, it does not maintain the target identity, and consequently, requires an online clustering method to detect the peaks of the particle weights and applies data association to each cluster.

Numerous online multi-object tracking methods deal with each tracker independently [1, 14, 22–24]. In [1], a method based on a particle filter and two human detectors with different features was developed, where the observation model depends on the associated detection, the detector confidence density and the likelihood of appearance. In addition, Shu *et al.* [14] introduced a part-based pedestrian detector for online multi-person

[*] Corresponding author

Email addresses: m_naiel@ece.concordia.ca (Mohamed A. Naiel), omair@ece.concordia.ca (M. Omair Ahmad), swamy@ece.concordia.ca (M.N.S. Swamy), jlim@hanyang.ac.kr (Jongwoo Lim), mhyang@ucmerced.edu (Ming-Hsuan Yang)

Table 1: Representative online multi-object tracking algorithms. AM: appearance model, J/I: joint/independent, MU: model update, PF: particle filter, KF: Kalman filter, CVM: constant velocity motion model, MCMC: Markov Chain Monte Carlo, CH: color histogram, LBP: local binary patterns, BOW: bag of words, DCD: detection confidence density, SS: sample selection, PGF: probabilistic gating function, $q(.)$: proposal distribution, SGM: sparsity-based generative model, PGM: 2DPCA-based generative model, SDC: sparsity-based discriminative classifier.

Algorithm	Search and proposal distribution	J/I	Sample descriptor	Data association	MU	Likelihood function
Okuma *et al.* [12]	Mixture particle filters, $q(.) \propto$ new observation and propagated particles	J	HSV CH	NA	No	Bhattacharyya similarity
Breitenstein *et al.* [1]	PF with CVM	I	RGI, LBP	Boosted classifier, PGF and position	Yes	Distance between each particle and the associated detection, DCD, and Boosted classifier
Yang *et al.* [22]	Bayesian filtering	I	RGB, shape, BOW	CVM, position and scale	Yes	Joint likelihood of AM features
Shu *et al.* [14]	Detector based or KF	I	CH, LBP	SVM classifier, position, size	Yes + SS	KF (if no associated detection to the tracker)
Zhang *et al.* [23]	Mean shift tracker or KF	I	CH, shift vector	Size, search area, tracker re-detection	Yes	Combination of Mean-shift and KF
Schumann *et al.* [24]	PF with random walk or CVM	I	RGB CH	Overlap ratio	Yes	Detector confidence
Duffner & Odobez [17]	MCMC with random walk, $q(.) \propto$ new detections and sampled particles	J	HSV CH	Overlap ratio and position, tracker re-detection	Yes	The product of the visible individual targets likelihoods
Proposed method	PF with detection based CVM, $q(.) \propto$ new associated detections and propagated particles	I	Grayscale	Overlap ratio, SGM and PGM, tracker re-detection	Yes + SS	SDC, different weights for newly created and propagated particles

tracking. This method combines the detection results with the Kalman filter, where data association is performed every frame, and the filter is used when occlusion occurs. Recently, Zhang *et al.* [23] used the mean-shift trackers and the Kalman filter for multi-person tracking, where trackers are either weakly or strongly trained. We note that these methods are likely to have low recall as the detector and tracker are not integrated within the same framework.

The degeneracy problem of particle filters [25] has been addressed in several methods [26–29] with more effective proposal distributions and re-sampling steps. Rui and Chen [27] used the unscented Kalman filter for generating the proposal distribution, and Han *et al.* [30] used a genetic algorithm to increase the diversity of the particles. Recently, the Metropolis Hastings algorithm has been used to sample particles from associated detections in the tracking-by-detection framework [29]. We note that the above-mentioned methods do not exploit the collaboration between detectors and trackers [27, 30], or do not consider the effect of false positive detections on the trackers [29].

An adaptive appearance model is one of the important factors for effective object tracking as it accounts for appearance change [31, 32]. In [12], the appearance model is fixed during the tracking process and thus, may result in tracking failure. On the other hand, the trackers are updated with positive samples [23] straightforwardly without differentiating whether they contain noise or not. As multiple objects are likely to be occluded, it is necessary to analyze the samples and reduce the likelihood of including noisy samples for model update. Recently, the appearance models [14] have been updated by the detected non-occluded object parts rather than the holistic samples.

In this paper, we propose an online multi-object tracking scheme by using a robust collaborative model for interaction between a number of single-object trackers with sparse representation-based discriminative classifiers [33, 34], and a pre-trained object detector in the particle filter framework, where every target is tracked independently to avoid the high computational complexity of the joint probability with increasing number of targets. A novel sample selection scheme is used to update each tracker by using key samples with high confidence from the trajectory of an object, where the key sample represents the association between the tracker and a detection at time, t. In addition, we present a data association method with partial occlusion handling by using diverse generative models composed of sparsity-based generative model [34], and two-dimensional principal component analysis (2DPCA) [35] generative model. Finally, we introduce a 2DPCA generative model to re-identify lost targets. Experimental results on benchmark datasets demonstrate that the proposed scheme generally outperforms state-of-the-art methods.

2. Overview of the proposed scheme

The proposed multi-object tracking scheme consists of three main components: a pre-trained object detector, a data association module and a number of single-object trackers. Figure 1 shows the block diagram of the proposed scheme, wherein only one single-object tracker is shown. The object detector is applied on every frame and supports the data association module with a set of detections $\mathcal{D}^t$ at time t. The object tracker adopts a hybrid motion model, and a particle filter with a robust collaborative model is used to estimate the target location. The appearance model consists of a sparsity-based discriminative classifier (SDC) with holistic features, a sparsity-based generative model (SGM) with local features, and a 2DPCA-based generative model (PGM) with holistic features. The SDC is used

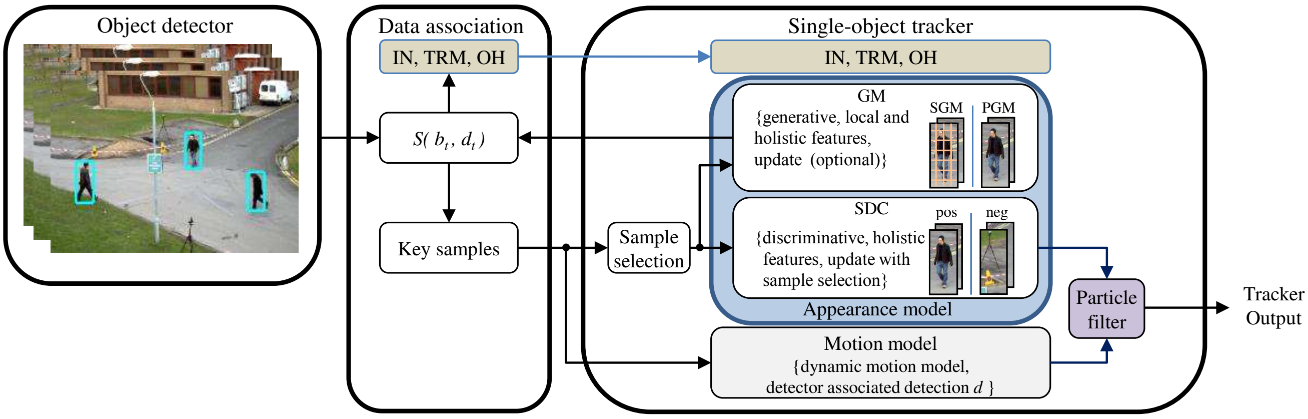


Figure 1: Block diagram of the proposed multi-object tracking scheme, where IN, TRM, OH, pos, and neg denote initialization, termination, on-hold, positive, and negative, respectively (see text for details).

to compute each sample confidence score of the particle filter, while the SGM and PGM are used to solve the data association problem. Each tracker also contains a sample selection scheme to update the appearance model with high confidence key samples. Finally, the data association module is used to construct the similarity matrix S to match detections, $d_t \in \mathcal{D}^t$, with existing trackers, $b_t \in \mathcal{B}_e^t$, at time t. Furthermore, it determines initialization, termination and on-hold states of the trackers, and supports the tracker with key samples from the target trajectory.

In this paper, we used the fast pedestrian detector (FPD) [36] for multi-person tracking. In Section 4, we used other pre-trained detectors, such as the on-road vehicle detector proposed in [37], and the method in [38] to measure the tracking performance on several detection conditions and different types of objects.

3. Tracking scheme

Each object tracker is based on the particle filter tracking framework that uses the sparse representations and 2DPCA as the appearance model. We incorporate two measurements from the detector and tracker into the particle filter, and propose a novel collaborative model that directly affects the likelihood function to obtain the posterior estimate of the target location. We construct the appearance model of the target by using discriminative and generative appearance models, for the likelihood function and the data association. In the following, we use a gate function $\mathcal{I}_{b_t}$ to represent the state of the tracker b_t when associated to the detection d_t at time t. The gate function is defined as

$$\mathcal{I}_{b_t} = \begin{cases} 1, & \text{if } b_t \text{ is associated with } d_t \text{ at time } t \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

3.1. Particle filter

In the Bayesian tracking framework, the posterior at time t is approximated by a weighted sample set $\{\mathbf{x}_t^i, \boldsymbol{w}_t^i\}_{i=1}^{N_s}$, where $\boldsymbol{w}_t^i$ is the weight of particle, $\mathbf{x}_t^i$, and N_s is the total number of particles. The state $\mathbf{x}$ consists of translation (x, y), average velocity (v_x, v_y), scale $\hat{s}$, rotation angle θ, aspect ratio η, and skew direction ϕ.

The measurement model of the proposed particle filter consists of two types. The first measurement is available every time t from the propagated particles $\mathbf{z}_{1:t}$. The second measurement is from the newly created particles that are available at time t when a detection window, d_t, is associated to a tracker, b_t (i.e., $\mathcal{I}_{b_t} = 1$). Assume that at time t, the tracker b_t is associated to a detection d_t, then we sample candidate particles from the importance density, $q(\mathbf{x}_t^i|\mathbf{x}_{1:t-1}^i, \mathbf{z}_{1:t}, d_t)$. The posterior probability of the candidate location given the available measurements, $p(\mathbf{x}_t|\mathbf{z}_{1:t}, d_t)$, is

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}, d_t) \approx \sum_{i=1}^{N_s} \boldsymbol{w}_t^i \delta(\mathbf{x}_{1:t} - \mathbf{x}_{1:t}^i) \tag{2}$$

where

$$\boldsymbol{w}_t^i \propto \boldsymbol{w}_{t-1}^i \frac{p(\mathbf{z}_{1:t}, d_t|\mathbf{x}_t^i) p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{q(\mathbf{x}_t^i|\mathbf{x}_{1:t-1}^i, \mathbf{z}_{1:t}, d_t)} \tag{3}$$

and $p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$ is the transition probability. In the proposed method, the particles are resampled every time t, and then we have $\boldsymbol{w}_{t-1}^i = 1/N_s, \forall i$, and we ignore $\boldsymbol{w}_{t-1}^i$ term. In the current frame, since the propagated particles sampled at time t corresponding to the tracker position in the previous frame and the particles sampled at time t from the associated detection are independent, the particle weights are computed by

$$\boldsymbol{w}_t^i \propto \frac{p(\mathbf{z}_{1:t}|\mathbf{x}_t^i) p(d_t|\mathbf{x}_t^i) p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)}{q(\mathbf{x}_t^i|\mathbf{x}_{1:t-1}^i, \mathbf{z}_{1:t}, d_t)} \tag{4}$$

where $p(d_t|\mathbf{x}_t^i)$ is the likelihood of detection d_t given the candidate location $\mathbf{x}_t^i$. To determine the term $p(d_t|\mathbf{x}_t^i)$, the confidence value of the object detector is required at every candidate location from each tracker, which is computationally expensive. Thus, we simplify this detection likelihood term using the Bayes' rule as

$$p(d_t|\mathbf{x}_t^i) = p(\mathbf{x}_t^i|d_t) p(d_t) / p(\mathbf{x}_t^i) \propto p(\mathbf{x}_t^i|d_t) \tag{5}$$

where $p(\mathbf{x}_t^i|d_t)$ represents the probability of the candidate location given that the tracker is associated to a detection, d_t. Let the proposal distribution be defined as

$$q(\mathbf{x}_t^i|\mathbf{x}_{1:t-1}^i, \mathbf{z}_{1:t}, d_t) \propto p(\mathbf{x}_t^i|d_t)p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i) \tag{6}$$

By substituting into (4) with the simplification from (5) and (6), the simplified particle weights with Markovian model can be computed by

$$\boldsymbol{w}_t^i \propto p(\mathbf{z}_t|\mathbf{x}_t^i) \tag{7}$$

By normalizing the particle weights, the resulting state estimate is represented as a weighted average of the candidate locations. This makes the proposed scheme more robust to noisy detection results compared to maximum a posteriori methods.

When there is no detection associated to a tracker (i.e., $\mathcal{I}_{b_t} = 0$), the proposed particle filter can be simplified to the bootstrap particle filter [25]. In the bootstrap particle filter, the measurement model consists of the tracker measurements $\mathbf{z}_{1:t}$ and the importance density at time t can be defined as $q(\mathbf{x}_t^i|\mathbf{x}_{1:t-1}^i, \mathbf{z}_{1:t}) \propto p(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i)$. It can be shown that the particle weights can be represented by (7).

3.1.1. *Motion model*

In the proposed method, we adopt a hybrid motion model based on the first-order Markov chain and the associated detection. The new candidate state $\mathbf{x}_t^d$ at time t is provided to the motion model if a detection is successfully associated to the tracker (i.e., $\mathcal{I}_{b_t} = 1$), and the initial velocity is set to be the average velocity of the tracker particles. The candidate state at time t, $\mathbf{x}_t$, relates to the set of propagated particles X^{b_t} and the set of associated detection X^{b_t,d_t} by

$$\mathbf{x}_t = \begin{cases} F\mathbf{x}_{t-1} + Q & \text{if } \mathbf{x}_t \in X^{b_t} \\ \mathbf{x}_t^d + P & \text{if } \mathbf{x}_t \in X^{b_t,d_t} \end{cases} \tag{8}$$

where Q and P are the Gaussian noise vectors, $N_s = N_s^P + N_s^\Gamma$, and N_s^P and N_s^Γ are the cardinality of X^{b_t} and X^{b_t,d_t}, respectively. In the above equation, F denotes the transition matrix of size 8×8, which is defined as

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

3.1.2. *Robust collaborative model*

The object detector applies computationally expensive space-scale search to the entire image to localize specific class of objects, and proposes candidate locations that have high probability of existence. To exploit high confidence associated detections, we incorporate a set of new particles, X^{b_t,d_t}, in the likelihood function, to allow the object detector to guide the trackers. Let $H(\mathbf{x}_t^i)$ denote SDC tracker confidence score of candidate $\mathbf{x}_t^i$. The likelihood of the measurement, $\mathbf{z}_t$, can be computed by

$$p(\mathbf{z}_t|\mathbf{x}_t^i) = \pi^i H(\mathbf{x}_t^i) \tag{10}$$

where

$$\pi^i = \begin{cases} 1-\gamma & \text{if } \mathcal{I}_{b_t} = 1, \mathbf{x}_t^i \in X^{b_t} \\ \gamma & \text{if } \mathcal{I}_{b_t} = 1, \mathbf{x}_t^i \in X^{b_t,d_t} \\ 1 & \text{otherwise, i.e., } \mathcal{I}_{b_t} = 0 \end{cases} \tag{11}$$

and $\gamma \in [0, 1]$ is the collaborative factor. In (10), the particles from the associated detections and previously propagated particles are weighted differently. Figure 2 shows the effect of changing the collaborative factor value. Figure 3(a) and (b) show an example of particle weights for the detector particles and the propagated particles using $\gamma = 0.54$. If $\mathcal{I}_{b_t} = 1$ and $\gamma > 0.5$, the weight π^i allows the detector to guide the tracker by giving more weights to the newly associated particles than the propagated particles. However, a detector may have false positives, and thus, the tracker should not depend completely on the detector. From our experiments, we find that the proposed scheme with the value of γ between 0.5 and 0.85 performs best. If the detector suffers from missing detections (i.e., $\mathcal{I}_{b_t} = 0$), the likelihood function in (10) will only depend on the previously propagated particles $\mathbf{x}_t^i \in X^{b_t}$, which represent the bootstrap particle filter [25]. Our collaborative model is based on the hybrid motion model that incorporates associated detections with object dynamics. In contrast, the motion model adopted in [1] depends only on propagated particles, and the likelihood function depends on tracker appearance model and the detector confidence density. The collaborative model in [12] only exists in the proposal distribution and the likelihood is without weighting collaborative factor.

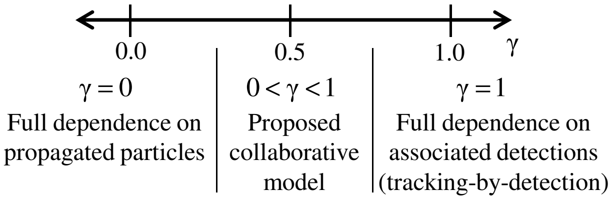
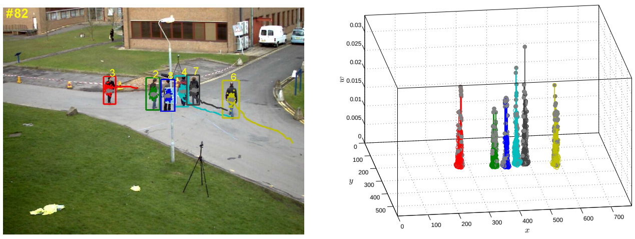

0.0
0.5
1.0
γ
γ = 0
Full dependence on propagated particles
0 < γ < 1
Proposed collaborative model
γ = 1
Full dependence on associated detections (tracking-by-detection)

Figure 2: Effect of changing the collaborative factor γ.



(a)

(b)

Figure 3: Effect of the proposed collaborative model on the tracker particles. (a) Illustrates the candidate particles proposed by the object detector (masked as gray) and propagated particles (colored). (b) Particles weights for new (masked as gray) and propagated particles (colored).

3.1.3. Resampling

In each frame, the set of candidate particles $\{\mathbf{x}_t^i, \boldsymbol{w}_t^i\}_{i=1}^{N_s}$ are resampled to avoid the degeneracy problem. The resampling process also allows the detector to guide the tracker effectively. As each tracker resamples particles based on particle weights computed from the proposed collaborative model (10), the propagated particles with low weights are replaced with newly created particles from the associated detections.

3.2. Appearance model

In the proposed method, the SGM and SDC are used in a way different from that in [34]. First, we do not use the collaboration between SGM and SDC [34], instead we use SGM with PGM to compute the similarity matrix of the data association module for occlusion handling (23), and the modified SDC model is used to compute the likelihood of the particle filter (10). The number of particles in the filter is usually larger than the number of detections and trackers at time t, and the computational complexity of SDC is lower than SGM. Therefore, the resulting tracker is more efficient. Second, our SDC uses the downsampled grayscale image without the feature selection method used in [34]. Third, our SDC confidence measure depends on the sparsity concentration index [33]. Finally, we propose the key sample selection scheme to update the appearance models with high confidence samples.

3.2.1. Sparsity-based discriminative classifier

We construct a discriminative sparse appearance model to compute the confidence score as used in (10). The initial training samples are collected in a similar way to [34], where each SDC tracker is initialized using N_p positive samples drawn from the object center with a small variation from the center of the detection state $\mathbf{x}_t^d$, and N_n negative samples are taken from the annular region surrounding the target center without overlap with a detection window d_t. Next, each sample is normalized to a canonical size of $(m \times n)$, and vectorized to be one column of the matrix $A \in \mathbb{R}^{r \times N^t}$, where $r = mn$ and $N^t = N_p + N_n + N_{p,u}^t + N_{n,u}^t$, such that $N_{p,u}^t$ and $N_{n,u}^t$ denote the buffer size of the selected key samples up to time t. Let the measurement corresponding to the candidate location $\mathbf{x}_t^i$ be denoted by $\mathbf{z}_t^i \in \mathbb{R}^r$. We obtain the sparse coefficients α^i for the i^{th} candidate by solving the following optimization problem,

$$\min_{\alpha^i} \left\|\mathbf{z}_t^i - A\alpha^i\right\|_2^2 + \lambda_{SDC} \left\|\alpha^i\right\|_1 \tag{12}$$

We compute the classifier confidence score by

$$H(\mathbf{x}_t^i) = \exp\left(-\frac{(\varepsilon_+^i - \varepsilon_-^i)}{\sigma}\right)\Omega(\alpha^i) \tag{13}$$

where $\varepsilon_+^i = \left\|\mathbf{z}_t^i - A_+\alpha_+^i\right\|_2^2$ is the reconstruction error of the candidate $\mathbf{z}_t^i$ with respect to the template set of the positive class A_+, and the sparse coefficient vector of the i^{th} candidate that corresponds to the positive class, α_+^i. Similarly, $\varepsilon_-^i = \left\|\mathbf{z}_t^i - A_-\alpha_-^i\right\|_2^2$ is the reconstruction error of the same candidate $\mathbf{z}_t^i$ with respect to the template set of the negative class A_-, and the corresponding sparse coefficient vector α_-^i. The parameter σ adjusts the confidence measure, and $\Omega(\alpha^i)$ represents the sparsity concentration index (SCI) [33] defined as

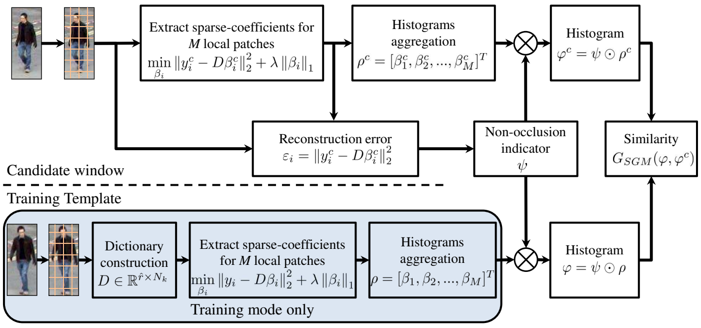

Extract sparse-coefficients for M local patches
$\min_{\beta_i} \|y_i^c - D\beta_i^c\|_2^2 + \lambda \|\beta_i\|_1$
Histograms aggregation
$\rho^c = [\beta_1^c, \beta_2^c, ..., \beta_M^c]^T$
Histogram
$\varphi^c = \psi \odot \rho^c$
Reconstruction error
$\varepsilon_i = \|y_i^c - D\beta_i^c\|_2^2$
Non-occlusion indicator
ψ
Similarity
$G_{SGM}(\varphi, \varphi^c)$
Candidate window
Training Template
Dictionary construction
$D \in \mathbb{R}^{\hat{r} \times N_k}$
Extract sparse-coefficients for M local patches
$\min_{\beta_i} \|y_i - D\beta_i\|_2^2 + \lambda \|\beta_i\|_1$
Histograms aggregation
$\rho = [\beta_1, \beta_2, ..., \beta_M]^T$
Histogram
$\varphi = \psi \odot \rho$
Training mode only

Figure 4: Block diagram of the sparsity-based generative model.

$$\Omega(\alpha^i) = \frac{J \cdot \max_j \|\delta_j^{'}(\alpha^i)\|_1 / \|\alpha^i\|_1 - 1}{J - 1} \in [0, 1] \tag{14}$$

where $\delta_j^{'}$ is a function that selects the coefficients corresponding to the j^{th} class and suppresses the rest, and J is the number of classes ($J = 2$ in this work). The SCI checks the validity of a candidate such that it can be represented by a linear combination of the training samples in one class. When the sparse coefficients concentrate in a certain class, the SCI value is high. This index allows each tracker to assign high weights to candidates resembling the positive training samples, and rejects others related to other targets or background structures.

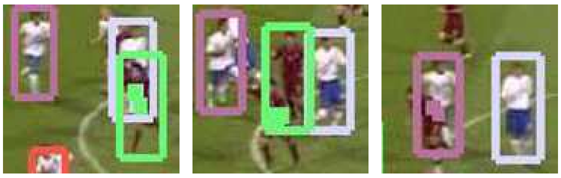
The SDC tracker is updated every R_u frames using the selected key samples, K_u^t (Section 3.3). At each key sample location, we collect positive and negative samples as part of the initialization process. To leverage between computational load and memory requirement, we set the maximum number of positive and negative samples. If the number of positive, $N_{p,u}^t$ or negative, $N_{n,u}^t$ samples exceeds the limit, we replace the old samples (other than those collected in the first frame) with the new selected key samples.

3.2.2. Sparsity-based generative model

We use a sparsity-based generative model to measure similarity in the data association module. Figure 4 illustrates the block diagram of the proposed SGM in the training and test modes. The training template consists of M local patches, $\{y_i\}_{i=1}^M$ and each patch of size $\hat{m} \times \hat{n}$. These M patches are vectorized[1] and quantized into N_k centroids using the k-means algorithm to construct the dictionary $D \in \mathbb{R}^{\hat{r} \times N_k}$ ($\hat{r} = \hat{m}\hat{n}$). For the i^{th} patch, y_i, the sparse-coefficients, $\beta_i \in \mathbb{R}^{N_k \times 1}$, is computed by

$$\min_{\beta_i} \|y_i - D\beta_i\|_2^2 + \lambda_{SGM} \|\beta_i\|_1 \tag{15}$$

[1] The vectorization function is defined as Mat2Vec: $\mathbb{R}^{m \times n} \to \mathbb{R}^r$, where $r = mn$ is the dimension of the vector, and $(m \times n)$ is the order of the input matrix. The inverse of the vectorization function is defined as Vec2Mat: $\mathbb{R}^r \to \mathbb{R}^{m \times n}$.



(a) (b) (c)

Figure 5: Sample results for SGM partial occlusion handling scheme, where the marked patches with the same tracker color are the patches at which SGM reconstruction error is greater than the SGM error threshold.

The adopted SGM is concerned with representing the appearance of the positive class of the tracker by using the sparse coefficients of M local patches of the object and candidate location c, where each location is represented by a sparse histogram feature vector $\rho = [\beta_1, \beta_2, \dots, \beta_M]^T$, and $\rho^c = [\beta_1^c, \beta_2^c, \dots, \beta_M^c]^T$, corresponding to the initial object and the candidate location, respectively. To handle occlusion, the patch reconstruction error, $\{\varepsilon_i = \|y_i^c - D\beta_i^c\|_2^2\}_{i=1}^M$, is used to suppress the coefficients of occluded patches. Let ψ_i be the non-occlusion indicator for the i^{th} patch and is computed by

$$\psi_i = \begin{cases} 1_{N_k,1} & \text{if } \varepsilon_i < \varepsilon_0 \\ 0_{N_k,1} & \text{otherwise} \end{cases} \tag{16}$$

where $1_{N_k,1}$, and $0_{N_k,1}$ denote the vector of size N_k of ones and zeros. The final histogram can be represented by $\varphi = \psi \odot \rho$, and $\varphi^c = \psi \odot \rho^c$, corresponding to the training template, and the candidate location, where $\odot$ denotes the element-wise multiplication. By taking the spatial representation into consideration, the resulting histogram, φ can handle occlusion effectively. Figure 5 illustrates the effect of the partial occlusion handling scheme. If the reconstruction error is greater than the threshold, ε_0, then the non-occlusion indicator, ψ, suppresses these patches. The generative model similarity, $G_{SGM}(b_t, c)$, between the candidate φ_c and the model φ is measured by using the intersection kernel.

As in [34], the dictionary, D, is fixed during the tracking process, while the sparse histogram of the initial template, $\rho_{initial}$, is updated every update rate, R_u. The sparse histogram is updated by

$$\rho_{new} = \mu\rho_{initial} + (1-\mu)\rho_K \tag{17}$$

where μ is the learning rate, and ρ_K represents the sparse histogram corresponding to the selected key sample from the set K_u^t that provides the maximum similarity to the training templates (see Section 3.3 for the sample selection scheme). This conservative update scheme by using the confidence key samples and maintaining the initial template provide effective tracking.

3.2.3. 2DPCA-based generative model

In addition to part-based SGM, we use a holistic generative model based on the 2DPCA scheme [35], referred to as PGM, to solve the data association problem. The reason being that a combination of PGM and SGM increases the tracking performance (see Section 4). For each tracker b_t, we use N positive samples, $\{Y_j\}_{j=1}^N$ each of size $m \times n$, where samples are taken from the positive class of the initial target location, or selected key samples, K_u^t. Each j^{th} sample Y_j is projected by the orthonormal matrix $V \in \mathbb{R}^{n \times r_1}$, $r_1 \leq n$ and form $F^j = Y_j V$, of size $m \times r_1$. The image covariance matrix Γ is defined by

$$\Gamma = \frac{1}{N}\sum_{j=1}^{N}(Y_j - \bar{Y})^\top (Y_j - \bar{Y}) \tag{18}$$

where $\bar{Y}$ is the average image of all training samples, and Γ is the nonnegative definite matrix. The objective of 2DPCA is to find the optimal orthonormal matrix, V_{opt}, that maximizes the total scatter in the learned subspace. The total scatter criterion $\mathbf{J}(V)$ is defined by

$$\mathbf{J}(V) = V^\top \Gamma V \tag{19}$$

The optimal projection matrix V_{opt} is composed of the r_1 eigenvectors of matrix Γ corresponding to the first r_1 largest eigenvalues, where the vectors are stacked together in matrix V of size $n \times r_1$. We extract features of the j^{th} training example, Y_j, through projecting on matrix V, as $F^j = Y_j V$, and then we vectorize the resulting feature matrix and have the feature vector f^j of size $(1 \times mr_1)$.

For each candidate location, we project the candidate sample, Y^c, using the matrix V, and vectorize the resulting matrix to obtain the test feature vector f^c of size $1 \times mr_1$. The nearest neighbor classifier is used to infer the index of the j^{th} training example, $\hat{j}$ closest to the test vector f^c

$$\hat{j} \leftarrow \underset{j \in \{1,2,\dots,N\}}{\text{argmin}} \|f^c - f^j\|_2 \tag{20}$$

where $\|.\|_2$ denotes the l_2-norm. The reconstruction error between the test image and the training examples is $\varepsilon_{PGM} = \|a_{\hat{j}} - a_c\|_2$, where $a_{\hat{j}} = \text{Mat2Vec}(F^{\hat{j}} V^\top)$ and $a_c = \text{Mat2Vec}(F^c V^\top)$. The similarity between the test and training features is computed by

$$G_{PGM} = \exp(-\varepsilon_{PGM}/\hat{\sigma}^2) \tag{21}$$

Figure 6 shows a sample intermediate output from the proposed PGM scheme. The PGM is able to retrieve the closest training patches in 2DPCA feature subspace, which provides accurate similarity measures in (21).

Similar to SDC tracker, PGM is updated every R_u frames, by using the initial positive and the selected key samples at time t, where $N = N_p + N_{p,u}^t$. To update 2DPCA feature space, we used a batch learning technique. In this scheme, we update the optimal projection matrix, V_{opt}, and extract the feature vectors, $\{f^j\}_{j=1}^N$. While the incremental 2DPCA learning has been used in [39], we find that batch learning performs more efficiently than the incremental learning scheme, since we replace some samples every update rate with newly selected key samples.

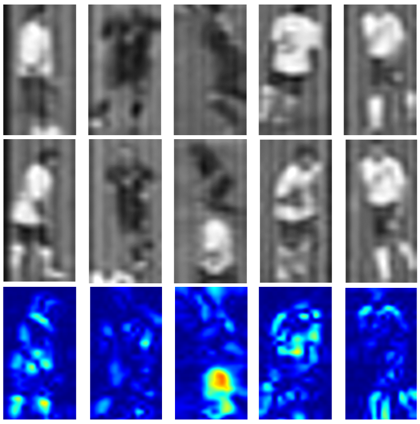


Figure 6: (Top) Reconstructed nearest neighbor training samples by PGM. (Middle) Reconstructed patches at candidate locations. (Bottom) Absolute reconstruction error, where the pixel with brighter color means high error value.

3.3. Sample selection

We propose a sample selection scheme to learn and adapt the appearance model for each tracker by using the samples with high confidence from the object trajectory, in a way similar to existing methods [1, 14, 40]. Examples for key sample locations in the object trajectory are shown in Figure 7, where two scenarios for the key samples are selected from the tracker history. The sample selection scheme alleviates the problem of including occluded samples for more effective model update and thus, reduces the drifting problem. The proposed sample selection scheme is based on the following criteria:

1. We measure the goodness of the key samples. A good key sample is one at which the tracker b_t does not intersect with other trackers or nearby detections except the associated detection d_t. We denote the set of good key samples at time t by K_g^t.

2. We use the online trained SDC tracker to measure the similarity between the current appearance model of the tracker, b_t, and the i^{th} good key sample $K_{g,i}^t \in K_g^t$ by

$$S_{DC}(b_t, K_{g,i}^t) = \exp(-(\varepsilon_+^i - \varepsilon_-^i)/\sigma^2) \tag{22}$$

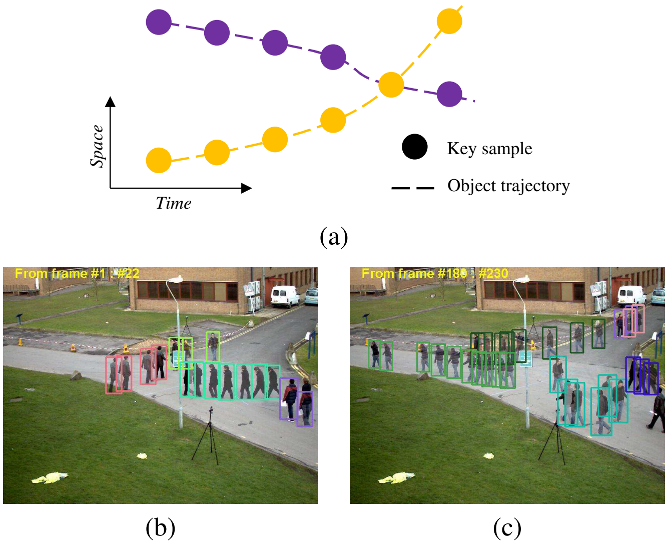
where $\varepsilon_+^i = \left\|\mathbf{z}_t^i - A_+\alpha_+^i\right\|_2^2$, $\varepsilon_-^i = \left\|\mathbf{z}_t^i - A_-\alpha_-^i\right\|_2^2$, and α_+^i and α_-^i are computed by using (12).

3. If $S_{DC}(b_t, K_{g,i}^t) > s_0 \geq 0$, where s_0 is the SDC similarity threshold, then this key sample is selected for the model update. The final set of selected key samples, K_u^t, which have high similarity with the SDC tracker, are used to update the tracker appearance model (Section 3.2). It should be observed that when $s_0 = 0$, all the samples are selected.

3.4. Data association

The similarity matrix S for data association measures the relation between a tracker $b_t \in \mathcal{B}_e^t$ and a detection $d_t \in \mathcal{D}^t$ by

$$S(b_t, d_t) = G(b_t, d_t)O(b_t, d_t) \tag{23}$$



(a)

(b)

(c)

Figure 7: (a) Key samples in the object trajectories and occlusion issues that should be handled, (b and c) Examples for key samples selected from object trajectories, using a sequence from the *PETS09-S2L1* dataset.

where $G(b_t, d_t) = G_{SGM}(b_t, d_t) + G_{PGM}(b_t, d_t)$ considers the appearance similarity between the tracker b_t and detection d_t, and $O(b_t, d_t)$ represents the overlap ratio between the tracker and the detection to suppress confusing detections, where the overlap ratio is based on the PASCAL VOC criterion [41].

The association is computed online by using the Hungarian algorithm to match a tracker to a detection in a way similar to existing methods [1, 14]. The proposed data association scheme iteratively finds the maximum in the matrix S, and associates the tracker b_t to a detection d_t if $S(b_t, d_t)$ is larger than a threshold s_1. The row and the column corresponding to $S(b_t, d_t)$ are removed. As the object detector is likely to miss some objects, using the similarity threshold, s_1, can alleviate the tracker to be updated with confusing nearby detections. Furthermore, we select a number of key samples to update the appearance model (Sections 3.3 and 3.2). We initialize new trackers with non-associated detection windows if the maximum overlap with other existing trackers is less than o_1 to avoid creating multiple trackers for the same target.

Re-detection module

A pre-trained object detector usually suffers from false positives and negatives, thereby causing trackers to drift. On the other hand, a tracker does not perform well in the presence of heavy occlusion or background clutters. To handle these challenging cases, we introduce the inactive or on-hold states before tracker termination in case the tracker misses a high number of detections.

Let the set of trackers on-hold be denoted as $\mathcal{B}_h^t$. When the tracker does not estimate the target location at an inactive state, we adopt the PGM (Section 3.2) to measure the similarity between the tracker on-hold $b_t \in \mathcal{B}_h^t$ and the new candidate location. When the tracker is in the inactive state b_t^h, it still can be reinitialized after checking the similarity with the new un-associated detection, d_t^u by $S_h(b_t^h, d_t^u) = G_{PGM}(b_t^h, d_t^u)$ (where G_{PGM} is computed by (21)). The inactive tracker is reactivated if $S_h(b_t^h, d_t^u) > s_2$, where s_2 is a pre-defined threshold. During the inactive state, the proposed tracker can reidentify lost targets and discriminate among trackers using the 2DPCA feature space learned from selected key samples.

4. Experimental results

4.1. Datasets

We evaluate the tracking performance of the proposed algorithm using seven challenging sequences, namely, the *PETS09-S2L1, PETS09-S2L2* [42], *UCF Parking Lot (UCF-PL)* dataset [14], *Soccer* dataset [13], *Town Center* dataset [43], and *Urban* as well as *Sunny* sequences from *LISA 2010* dataset [44], and compare it with that of several state-of-the-art online MOT methods.

The *PETS09-S2L1* sequence consists of 799 frames of 768 × 576 pixels recorded at 7 frames per second with medium crowd density. The *PETS09-S2L2* sequence consists of 442 frames with the same resolution and frame rate as the *PETS09-S2L1* sequence, but it contains heavy crowd density and illumination changes. The target objects undergo scale changes, long-term occlusion, and with similar appearance. The ground truth (GT) data from [45, 46] and [47] are used for evaluating the tracking results on *PETS09-S2L1* and *PETS09-S2L2*, respectively. The *Soccer* sequence consists of 155 frames of 960 × 544 pixels recorded at 3 to 5 frames per second. The challenging factors of this sequence include heavy occlusion, sudden change of motion direction of players, high similarity among players of the same team, and scale changes. The GT data provided by [13] are used for evaluation. On the *PETS09-S2L1*, *PETS09-S2L2* and *Soccer* sequences, the FPD detector [36] is used as the baseline detector for the proposed tracking scheme.

The *UCF-PL* dataset consists of 998 frames of 1920 × 1080 pixels recorded at 29 frames per second with medium crowd density, long-term occlusion, and targets of similar appearance. On this dataset, the detection results of the part-based pedestrian detector proposed in [14] are used for evaluation based on the GT data provided by [48].

The *Town Center* dataset consists of 4500 frames of 1080 × 1920 pixels recorded at 25 frames per second. The dataset contains medium crowd density, heavy occlusion, and scale changes. In [43], two categories of GT annotations are provided based on the full body and head regions of pedestrians. On this dataset, the aggregated channel feature (ACF) detector proposed by Dollár *et al.* [38] is used for performance evaluation. In the case of the full body of pedestrians, it has been observed that the ACF detector does not perform well on this sequence as the false positive rate is high. To alleviate this problem, the first 500 frames of this sequence are used to collect hard-negative samples related to the background clutters, and the ACF detector is re-trained using both the *INRIA* dataset [49] and hard-negative samples. In case of tracking multiple people based on the head regions, the positive training examples provided in [43] and negative samples collected from the first 500 frames of this sequence are used to train the ACF detector.

The *Urban* and *Sunny* sequences from the *LISA 2010* dataset [44] contain car images of 704 × 480 collected at 30 frames per second from a camera mounted on a moving vehicle. The *Urban* sequence (300 frames) was captured from an urban area with a low traffic density on a cloudy day, while the *Sunny* sequence (300 frames) was captured from a highway with medium traffic density on a sunny day. The challenging factors of these sequences include the effect of camera vibration, illumination changes, and the targets' scale changes; the GT data are provided by [44]. The pre-trained vehicle detector proposed in [37] is used for evaluation on this dataset.

4.2. Qualitative results

In this section, we study the qualitative performance of the proposed tracking scheme using the datasets mentioned above. Figures 8 and 9 show some of the tracking results and videos are available at `https://youtu.be/1nAUnU596UE`.

PETS09-S2L1. Figure 8(a) shows the sample tracking results of the proposed scheme on the *PETS09-S2L1* sequence. The proposed method performs well despite several short-term occlusions, scale and pose changes. Furthermore, it should be mentioned that the pre-trained FPD detector [36] misses objects that are close to the camera or those located far from the camera.

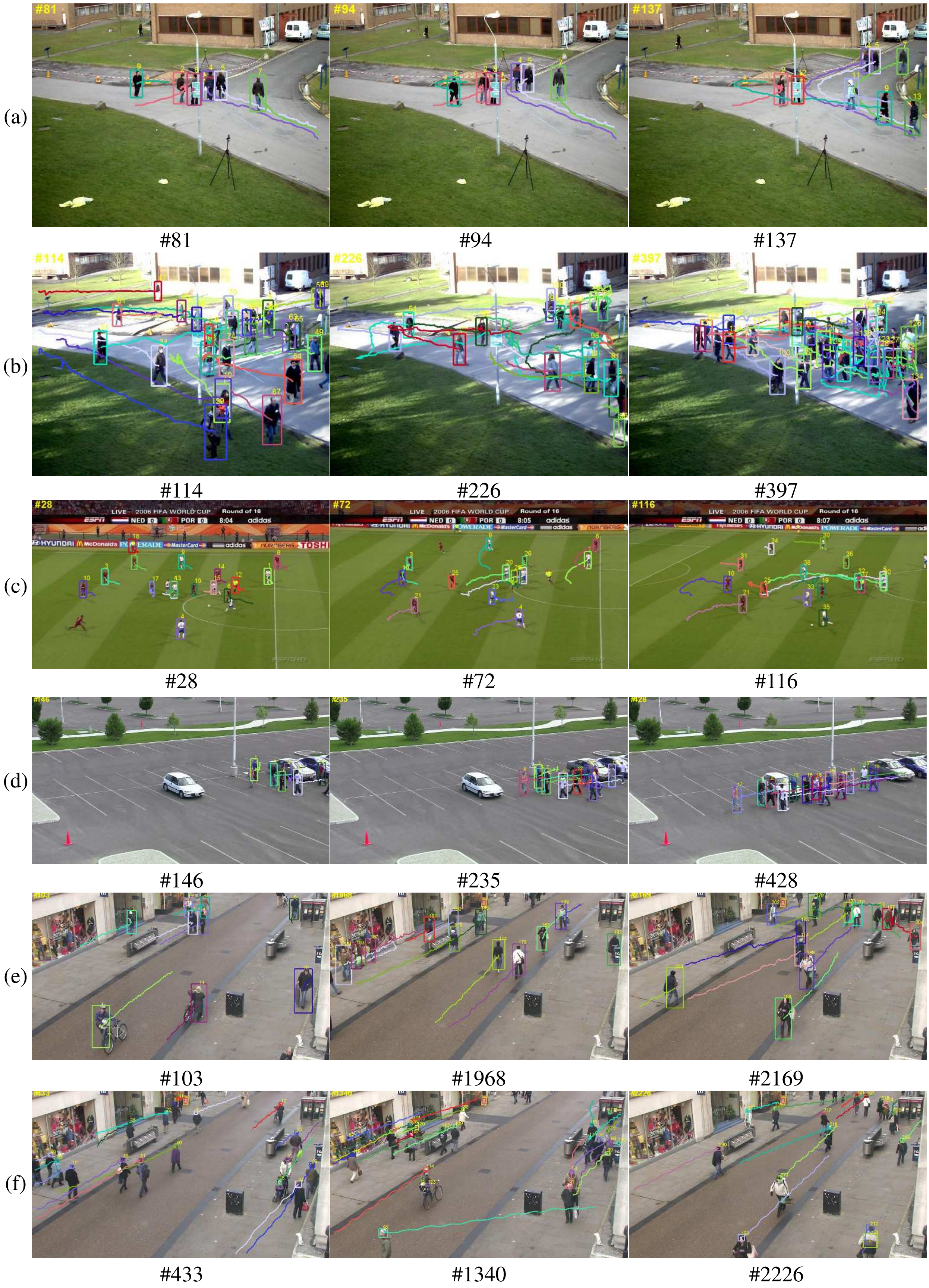
PETS09-S2L2. Figure 8(b) shows that non-occluded targets are tracked well although targets with long-term occlusions or located far from the camera are missed. Again, it should be mentioned that the FPD detector [36] misses numerous detections in this sequence due to the high crowd density.

Soccer. This sequence contains soccer players with similar visual appearance and fast motion. The FPD detector [36] is not trained to detect the soccer players at different poses. Nevertheless, the proposed scheme performs well with accurate short tracklets, as shown in Figure 8(c).

UCF-PL. This sequence contains crowds of medium density, with occlusions. Figure 8(d) shows some tracking results for the proposed scheme using the detector in [14]. Despite the challenges of the sequence, the proposed tracking scheme maintains long trajectories.

Town Center. The crowd density of this sequence is medium with a number of long-term occlusions. Figures 8(e) and (f) show sample tracking results corresponding to full body and head, respectively. While it is difficult to track the full human body due to heavy occlusions, or the head due to false positives, the proposed method performs well.

LISA 2010. Figures 9(a) and (b) show the sample results of our tracker using the detector in [37] on the *Urban* and *Sunny* sequences. The *Urban* sequence contains only one vehicle, but there is illumination change and the effect of camera vibrations. The *Sunny* sequence contains, on average, three non-occluded vehicles with different velocities. In spite of these challenges, the proposed scheme tracks the vehicles very well in both cases.



(a) #81 #94 #137

(b) #114 #226 #397

(c) #28 #72 #116

(d) #146 #235 #428

(e) #103 #1968 #2169

(f) #433 #1340 #2226

Figure 8: Sample tracking results for five sequences, the arrangement from top to bottom as (a) and (b) *PETS09-S2L1*, and *PETS09-S2L2*, respectively, (c) *Soccer* sequence, (d) *UCF-PL* sequence, (e) *Town Center* dataset (body), and (f) *Town Center* dataset (head).

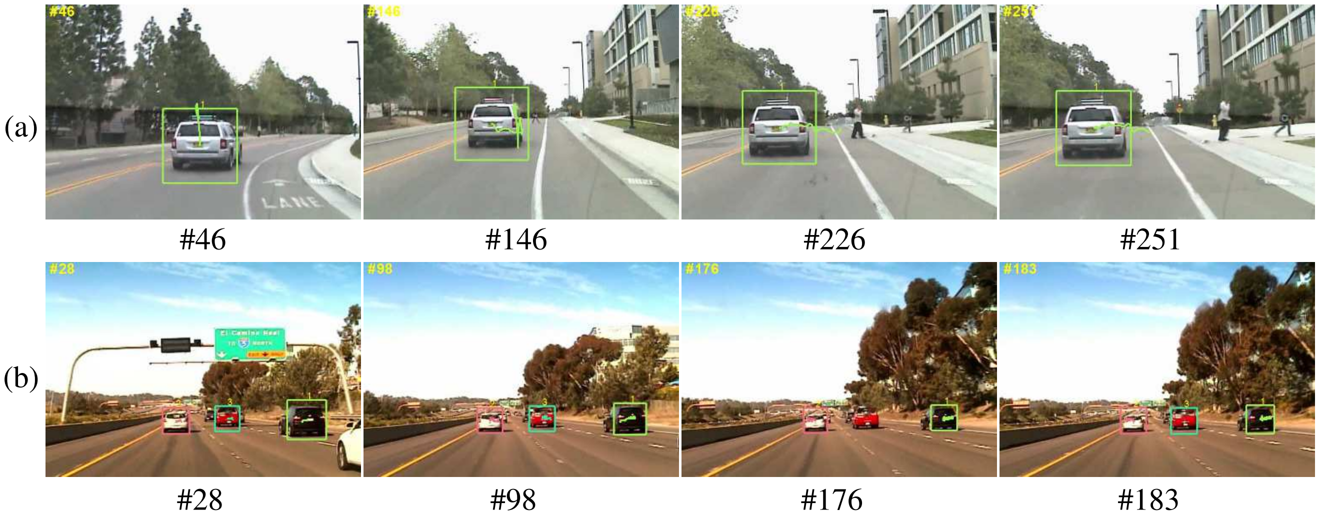


Figure 9: Sample tracking results for *LISA 2010* dataset, where (a) and (b) correspond to *Urban* and *Sunny* sequences, respectively.

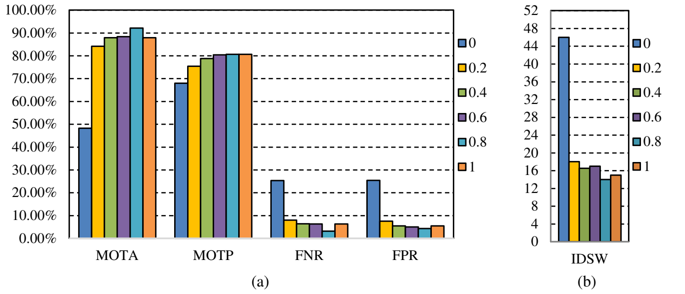


Figure 10: Performance of the proposed method on the *PETS09-S2L1* sequence for different values of the collaborative factor γ.

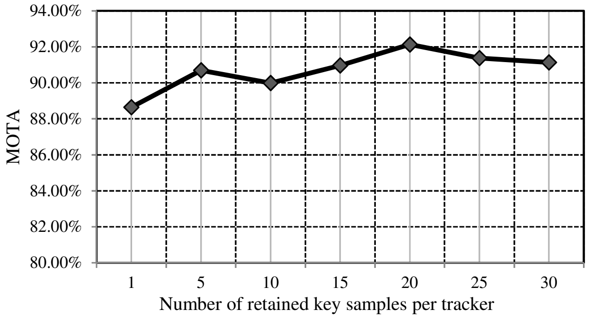


Figure 11: MOTA vs. number of retained key samples for the proposed tracker on the *PETS09-S2L1* sequence.

4.3. Quantitative results

We use the CLEAR MOT metrics [50] including multiple object tracking accuracy (MOTA), multiple object tracking precision (MOTP), false negative rate (FNR), false positive rate (FPR), and identity switches (IDSW) for evaluating the performance of the proposed tracker. We use the overlap threshold of 0.5 for all experiments. For this study, we set the various parameters to be $N_s^P = 150$, $N_s^\Gamma = 100$, $N_p = N_{p,u}^t = 10$, $N_n = N_{n,u}^t = 20$, $R_u = 10$, $\lambda_{SDC} = 0.02$, $\lambda_{SGM} = 0.01$, $\hat{\sigma} = 10^4$, $\varepsilon_0 = 0.8$, $\mu = 0.6$, $\hat{\sigma} = 5 \times 10^6$, $s_0 = 1.0$, $s_1 = 2.5$, $s_2 = 0.7$, and $o_1 = 0.2$. For the multi-person tracking sequences, namely, *PETS09-S2L1, PETS09-S2L2*, *UCF-PL*, *Soccer*, and *Town Center (Body)*, we use $m = 32$, $n = 16$, $M = 84$, $\hat{m} = \hat{n} = 6$ and $N_k = 50$. Further, for the multi-head tracking sequence, namely, *Town Center (Head)*, as well as the multi-vehicle tracking sequences, namely, *Urban* and *Sunny*, we use $m = n = 16$, $M = 16$, $\hat{m} = \hat{n} = 6$ and $N_k = 16$.

Effect of the collaborative factor. To measure the effect of the proposed collaborative model, we changed the value of the collaborative factor γ in the interval $[0, 1]$ in increments of 0.2. Figure 10 shows the performance of the proposed method with different values of γ for the *PETS09-S2L1* sequence. When $\gamma = 0$, the likelihood function of the particle filter is based completely on the propagated particles, and the proposed method does not perform well due to the degeneracy problem. When $\gamma = 1$, the likelihood function is based on the associated detections, and the tracker does not perform well due to false positives and missed detections. The proposed method performs best for this sequence when $\gamma = 0.8$, as can be seen from Figure 10. It is worth noting that for high tracking performance, the value of γ should be adjusted according to the detector used. For detectors with high precision and recall (the ones used in the *PETS09-S2L1*, *UCF-PL*, *Town Center (Head)*, *Urban* and *Sunny* sequences), the proposed tracker provides a high MOTA value when γ is in the interval of $[0.65, 0.85]$. On the other hand, when the detector has low precision and recall (the ones used in the case of *PETS09-S2L2*, *Soccer* and *Town Center (Body)* sequences), the proposed tracker provides a high MOTA value when γ is in the interval of $[0.5, 0.6]$.

Number of key samples. We analyze the effect of the number of key samples retained on MOTA using the *PETS09-S2L1* sequence. The appearance model (SDC, SGM, and PGM) is updated online at an update rate R_u of 10. Figure 11 shows the performance of the proposed tracker when the number of key samples retained is varied. We choose the number of retained key samples to be 20 at which the highest MOTA performance is exhibited, as seen from Figure 11.

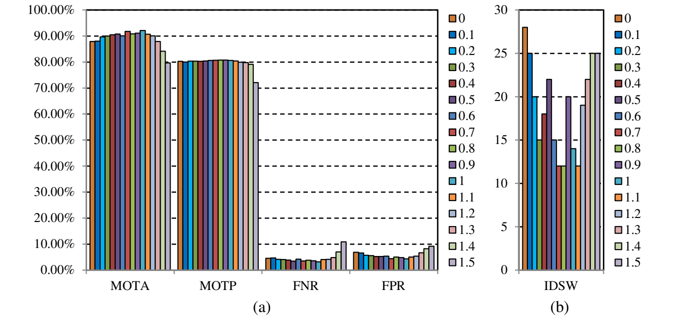


Figure 12: Performance of the proposed tracking scheme with respect to the SDC similarity threshold, s_0, using the *PETS09-S2L1* sequence.

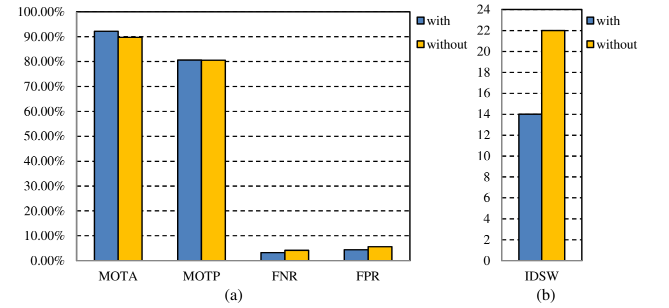


Figure 13: Performance of the proposed method with and without tracker re-detection on the *PETS09-S2L1* sequence.

Key sample selection. To demonstrate the strength of the proposed sample selection scheme, we examine the performance of the proposed tracking scheme by varying the SDC similarity threshold, s_0, from 0 to 1.5 in increments of 0.1. Figure 12 shows the performance of the proposed scheme at different SDC tracker similarity threshold values. When $s_0 = 1$, the proposed tracker exhibits the best performance in terms of MOTA. If $0 \leq s_0 < 1$, the performance is not as good in view of the fact that only a few or none of the key samples are rejected, and hence, occluded samples are likely to be selected. When $s_0 > 1.2$, the proposed tracker performs worse than that at $s_0 = 1$, since a large number of key samples are rejected. As such, we choose $s_0 = 1.0$ for all the experiments.

Effect of tracker re-detection. We analyze the effect of using the re-detection module on MOT tracking. Figure 13 shows that the proposed method with tracker re-detection scheme achieves slightly lower FNR and FPR than that obtained without using the tracker re-detection scheme, while maintaining approximately the same performance in terms of MOTA and MOTP values. The tracker re-detection scheme aims to reduce the number of identity switches and maintains long trajectories, without reducing the tracking performance.

Generative appearance models. We study the tracking performance of the proposed method by using several types of generative models to solve the data association problem in (23). These generative models are (1) SGM, as outlined in Section 3.2.2, which is based on local patch features (by substituting in (23) by $G = G_{SGM}$); (2) 2DPCA generative model, as proposed in Section 3.2.3, which is based on holistic features (by substituting in (23) by $G = G_{PGM}$); (3) combination of SGM and 2DPCA generative models as mentioned in Section 3.4; (4) principal component analysis (PCA)[2] generative model (instead of using the 2DPCA generative model); and (5) combination of SGM and PCA generative models.

The main differences between 2DPCA versus PCA are as follows. The covariance matrix in the case of 2DPCA can be computed directly from the image samples in 2D matrices rather than 1D vectors as in the case of PCA [35, 52]. The complexity for computing the covariance matrix using a 2DPCA-based appearance model is $\mathcal{O}(mn^2N)$, whereas the corresponding complexity using a PCA-based appearance model is $\mathcal{O}(m^2n^2N)$, when a set of N image samples, each of size $m \times n$ pixels, is used. Further, it may be pointed out that 2DPCA encodes the relationship among neighboring rows in a given set of image samples [52]. Such a relationship should have a positive effect on the tracking performance.

Table 2 shows the results on the seven sequences. Overall, the proposed scheme with SGM in conjunction with 2DPCA performs better than that by using SGM with PCA. In most sequences, the method of using SGM with 2DPCA or SGM with PCA performs better than that using only SGM. On a machine with 2.9 GHz CPU, the average tracking time per frame (over all the seven sequences without counting the time for object detection) for the proposed tracker with SGM and 2DPCA is 2.88 s whereas the corresponding time in the case of SGM and PCA is 2.90 s. Hence, this improvement in the performance of the proposed tracker is achieved without loss in speed.

4.4. Performance comparison

In this section, we evaluate the performance of the proposed algorithm with two online MOT methods in [53, 54] using the seven challenging sequences described in Section 4.1. Table 3 shows the performance of these two methods (using the original source code) along with that of the proposed tracker in terms of the various CLEAR MOT metrics. In addition, the performance of the proposed scheme is compared with the reported results of state-of-the-art online MOT methods [14, 23, 43, 55–58] using the sequences considered in these papers.

On the *PETS09-S2L1* and *PETS09-S2L2* sequences, the proposed scheme provides the second highest MOTA values. It also offers the highest and second highest MOTP values on the *PETS09-S2L1* and *PETS09-S2L2* sequences, respectively. This can be attributed to the proposed update mechanism, and the inactive or on-hold states of the tracker.

For the *Soccer* sequence, the proposed scheme performs better than the methods in [53, 54] despite fast camera motion and the presence of similar objects in the scenes. For the *UCF-PL* sequence, the MOTA value of the proposed method is higher than that of the methods in [14, 53, 54], using the same detector as in [14]. On the other hand, the MOTP value of the proposed

[2] The function `pcaApply` from toolbox [51] has been used to calculate the PCA.

Table 2: Performance of the proposed scheme using different generative models.

Sequence	Generative model	MOTA	MOTP	FNR	FPR	IDSW
PETS09-S2L1	SGM	89.08%	79.89%	5.11%	5.42%	17
	PCA	89.86%	79.97%	5.04%	4.76%	16
	SGM + PCA	90.12%	80.55%	5.34%	**4.30%**	**13**
	2DPCA	89.81%	79.82%	5.40%	4.41%	20
	Proposed	**92.13%**	**80.62%**	**3.19%**	4.33%	14
PETS09-S2L2	SGM	36.43%	71.19%	39.38%	26.31%	263
	PCA	45.69%	71.74%	35.92%	20.25%	**218**
	SGM + PCA	44.35%	71.54%	36.04%	21.30%	237
	2DPCA	46.06%	**71.77%**	36.59%	**19.33%**	221
	Proposed	**46.88%**	71.66%	**34.92%**	19.43%	258
Soccer	SGM	67.36%	70.28%	16.72%	14.49%	45
	PCA	70.33%	70.64%	18.85%	10.28%	38
	SGM + PCA	70.21%	**70.99%**	18.20%	11.03%	**36**
	2DPCA	71.13%	70.73%	17.00%	10.66%	49
	Proposed	**73.54%**	70.77%	**16.20%**	**9.45%**	38
UCF-PL	SGM	82.30%	71.84%	10.77%	6.27%	16
	PCA	82.14%	71.88%	10.44%	6.56%	21
	SGM + PCA	83.29%	71.81%	10.64%	**5.40%**	16
	2DPCA	81.89%	71.75%	11.47%	5.90%	18
	Proposed	**85.02%**	**71.89%**	**8.70%**	5.65%	**15**
Town Center (Body)	SGM	69.41%	73.82%	**17.08%**	12.81%	444
	PCA	70.19%	73.83%	18.18%	11.08%	351
	SGM + PCA	69.83%	73.89%	19.29%	10.37%	**320**
	2DPCA	**71.24%**	**74.02%**	18.02%	10.21%	337
	Proposed	70.16%	73.93%	19.35%	**9.95%**	342
Town Center (Head)	SGM	70.32%	68.86%	14.96%	14.48%	164
	PCA	72.15%	68.71%	14.25%	13.37%	163
	SGM + PCA	69.37%	68.78%	15.62%	14.77%	166
	2DPCA	70.43%	68.82%	15.06%	14.29%	**158**
	Proposed	**74.54%**	**69.15%**	**13.02%**	**12.21%**	**158**
LISA10 Urban	SGM	100.00%	82.67%	0.00%	0.00%	0
	PCA	100.00%	82.68%	0.00%	0.00%	0
	SGM + PCA	100.00%	82.68%	0.00%	0.00%	0
	2DPCA	100.00%	82.68%	0.00%	0.00%	0
	Proposed	100.00%	82.68%	0.00%	0.00%	0
LISA10 Sunny	SGM	97.22%	78.28%	0.78%	1.98%	0
	PCA	97.22%	78.28%	0.78%	1.98%	0
	SGM + PCA	97.22%	78.28%	0.78%	1.98%	0
	2DPCA	97.22%	78.28%	0.78%	1.98%	0
	Proposed	97.22%	78.28%	0.78%	1.98%	0
Average	SGM	76.51%	74.60%	13.10%	10.22%	-
	PCA	78.45%	74.72%	12.93%	8.53%	-
	SGM + PCA	78.05%	74.81%	13.24%	8.64%	-
	2DPCA	78.47%	74.73%	13.04%	8.35%	-
	Proposed	**79.94%**	**74.87%**	**12.02%**	**7.87%**	-

Note: The best and the second best results on each dataset are shown in boldface and underscored, respectively. The proposed method is SGM + 2DPCA.

technique is close to that of [14]. In addition, the proposed method has lower values for FNR and FPR than the methods in [14, 53, 54] do.

For the *Town Center* dataset, the proposed scheme is first evaluated to track the full body of pedestrians. In this case, the proposed scheme yields the second highest MOTP, FNR and FPR values compared to the methods in [14, 23, 43, 53, 54]. Next, the proposed scheme is evaluated on tracking the heads of pedestrians from the same dataset. The head regions in this sequence are less occluded than the full body, although the head detector has higher FPR than the full-body detector. As shown in Table 3, the proposed method performs well against other approaches [43, 53, 54, 58] in terms of MOTA. For the *Urban* and *Sunny* sequences from *LISA 2010* dataset, the proposed scheme provides a better performance than that provided by the methods in [53, 54] for tracking multiple vehicles on-road.

Table 3: Performance measures of CLEAR MOT metrics.

Sequence	Method	MOTA	MOTP	FNR	FPR	IDSW
PETS09-S2L1	Proposed	92.13%	**80.62%**	**3.19%**	**4.33%**	14
	Yoon *et al.* [53]*	66.64%	57.46%	17.99%	15.14%	34
	Bao and Yoon [54]*	89.94%	79.34%	4.83%	4.73%	23
	Zhang *et al.* [23]	**93.27%**	68.17%	-	-	19
	Zhou *et al.* [55]	87.21%	58.47%	-	-	-
	Breitenstein *et al.* [56]	79.70%	56.30%	-	-	-
	Gerónimo *et al.* [57]	51.10%	75.00%	45.20%	-	**0**
PETS09-S2L2	Proposed	46.88%	71.66%	**34.92%**	19.43%	258
	Yoon *et al.* [53]*	26.85%	47.99%	51.27%	28.86%	218
	Bao and Yoon [54]*	45.98%	**71.77%**	35.73%	**19.06%**	325
	Zhang *et al.* [23]	**66.72%**	58.21%	-	-	**215**
Soccer	Proposed	**73.54%**	**70.77%**	**16.20%**	**9.45%**	38
	Yoon *et al.* [53]*	29.99%	53.77%	52.89%	26.19%	**10**
	Bao and Yoon [54]*	54.25%	69.26%	35.45%	12.64%	24
UCF-PL	Proposed	**85.02%**	71.89%	**8.70%**	**5.65%**	**15**
	Yoon *et al.* [53]*	29.50%	45.33%	38.04%	33.95%	**15**
	Bao and Yoon [54]*	82.84%	73.33%	10.31%	6.49%	**15**
	Shu *et al.* [14]	79.30%	**74.10%**	18.30%	8.70%	-
Town Center (Body)	Proposed	70.16%	73.93%	19.35%	9.95%	342
	Yoon *et al.* [53]*	62.93%	48.66%	20.00%	17.14%	330
	Bao and Yoon [54]*	**79.07%**	73.46%	**11.19%**	**9.44%**	**307**
	Benfold and Reid [43]	61.30%	**80.30%**	21.00%	18.00%	-
	Zhang *et al.* [23]	73.61%	68.75%	-	-	421
	Shu *et al.* [14]	72.90%	71.30%	-	-	-
Town Center (Head)	Proposed	**74.54%**	69.15%	**13.02%**	12.21%	158
	Yoon *et al.* [53]*	73.90%	**70.16%**	17.23%	**9.49%**	**126**
	Bao and Yoon [54]*	70.65%	69.97%	16.31%	13.07%	320
	Poiesi *et al.* [58]	54.60%	63.70%	23.80%	21.70%	285
	Benfold and Reid [43]	45.40%	50.80%	29.00%	26.20%	-
LISA10 Urban	Proposed	**100.00%**	**82.68%**	**0.00%**	**0.00%**	**0**
	Yoon *et al.* [53]*	99.33%	81.98%	0.33%	0.33%	**0**
	Bao and Yoon [54]*	98.33%	82.52%	1.67%	**0.00%**	**0**
LISA10 Sunny	Proposed	**97.22%**	**78.28%**	**0.78%**	1.98%	**0**
	Yoon *et al.* [53]*	92.89%	77.20%	6.89%	**0.24%**	**0**
	Bao and Yoon [54]*	97.00%	77.83%	2.67%	0.34%	**0**

Note: $*$ denotes the results obtained by utilizing the code provided by the authors of the paper, where the detection results and GT annotations that have been used with the proposed scheme are used. The best and the second best results on each dataset are represented in boldface and underscored, respectively.

We note that the proposed scheme uses grayscale images as features, whereas the methods in [14, 23, 56] are based on the color or gradient information of the targets. In addition, the proposed scheme does not require the detector confidence density or a gate function in the data association step as in [1, 56], where the gate function provides higher weight for detections located in the direction of motion of the target.

5. Conclusion

In this paper, we have presented a robust collaborative model that enhances the interaction between a pre-trained object detector and a number of single-object online trackers in the particle filter framework. The proposed scheme is based on incorporating the associated detections with the motion model, in addition to the likelihood function providing different weights for the propagated and the newly created particles sampled from the associated detections, providing a reduction on the effect of the detector errors on the tracking process. We have exploited sparse representation and 2DPCA to construct diverse features that maximize the appearance variation among the trackers. Furthermore, we have presented a conservative sample selection scheme to update the appearance model of every tracker. Experimental results on benchmark datasets have shown that the proposed scheme outperforms state-of-the-art multi-object tracking methods in most of the cases.

Acknowledgments

The authors would like to thank Dr. Y. Wu for his helpful discussions and suggestions. They would also like to thank all the authors that made their codes available for comparison of the proposed algorithm with theirs and the anonymous reviewers for their constructive comments and suggestions. M.A. Naiel would like to acknowledge the support from Concordia University to conduct this research. This work is supported by research grants from the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Regroupement Stratégeique en Microsystémes du Québec (ReSMiQ) awarded to M.O. Ahmad and M.N.S. Swamy. J. Lim is supported by the National Research Foundation (NRF) of Korea grant #2014R1A1A2058501. M.-H. Yang is supported in part by the National Science Foundation (NSF) CAREER grant #1149783 and a gift from Panasonic.

References

[1] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. V. Gool, Robust tracking-by-detection using a detector confidence particle filter, in: Proc. Int. Conf. Comput. Vis., 2009, pp. 1515–1522.

[2] L. Leal-Taixé, G. Pons-Moll, B. Rosenhahn, Everybody needs somebody: Modeling social and grouping behavior on a linear programming multiple people tracker, in: Proc. Int. Conf. Comput. Vis. Workshops, 2011, pp. 120–127.

[3] W. Brendel, M. R. Amer, S. Todorovic, Multiobject tracking as maximum weight independent set, in: Proc. Comput. Vis. Pattern Recognit., 2011, pp. 1273–1280.

[4] A. Andriyenko, K. Schindler, Multi-target tracking by continuous energy mini-mization, in: Proc. Comput. Vis. Pattern Recognit., 2011, pp. 1265–1272.

[5] H. B. Shitrit, J. Berclaz, F. Fleuret, P. Fua, Tracking multiple people under global appearance constraints, in: Proc. Int. Conf. Comput. Vis., 2011, pp. 137–144.

[6] A. R. Zamir, A. Dehghan, M. Shah, GMCP-tracker: Global multi-object tracking using generalized minimum clique graphs, in: Proc. European Conf. Comput. Vis., 2012, pp. 343–356.

[7] A. Andriyenko, K. Schindler, S. Roth, Discrete-continuous optimization for multi-target tracking, in: Proc. Comput. Vis. Pattern Recognit., 2012, pp. 1926–1933.

[8] Z. Wu, A. Thangali, S. Sclaroff, M. Betke, Coupling detection and data association for multiple object tracking, in: Proc. Comput. Vis. Pattern Recognit., 2012, pp. 1948–1955.

[9] H. Izadinia, I. Saleemi, W. Li, M. Shah, (MP)^{2}T: Multiple people multiple parts tracker, in: Proc. European Conf. Comput. Vis., 2012, pp. 100–114.

[10] A. A. Butt, R. T. Collins, Multiple target tracking using frame triplets, in: Proc. Asian Conf. on Comput. Vis., 2012, pp. 163–176.

[11] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, K. Schindler, MOTChallenge 2015: Towards a benchmark for multi-target tracking, 2015, arXiv:1504.01942.

[12] K. Okuma, A. Taleghani, N. D. Freitas, J. J. Little, D. G. Lowe, A boosted particle filter: Multitarget detection and tracking, in: Proc. European Conf. Comput. Vis., 2004, pp. 28–39.

[13] Y. Wu, X. Tong, Y. Zhang, H. Lu, Boosted interactively distributed particle filter for automatic multi-object tracking, in: Proc. Int. Conf. on Image Process., 2008, pp. 1844–1847.

[14] G. Shu, A. Dehghan, O. Oreifej, E. Hand, M. Shah, Part-based multiple-person tracking with partial occlusion handling, in: Proc. Comput. Vis. Pattern Recognit., 2012, pp. 1815–1821.

[15] J. Vermaak, A. Doucet, P. Perez, Maintaining multimodality through mixture tracking, in: Proc. Int. Conf. Comput. Vis., 2003, pp. 1110–1116.

[16] Y. Jin, F. Mokhtarian, Variational particle filter for multi-object tracking, in: Proc. Int. Conf. Comput. Vis., 2007, pp. 1–8.

[17] S. Duffner, J. Odobez, Track creation and deletion framework for long-term online multi-face tracking, IEEE Trans. on Image Process. 22 (1) (2013) 272–285.

[18] E. Maggio, M. Taj, A. Cavallaro, Efficient multitarget visual tracking using random finite sets, IEEE Trans. on Circuits and Syst. for Video Technology 18 (8) (2008) 1016–1027.

[19] V. Eiselein, D. Arp, M. Patzold, T. Sikora, Real-time multi-human tracking using a probability hypothesis density filter and multiple detectors, in: Proc. IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance, 2012, pp. 325–330.

[20] R. Mahler, Multitarget Bayes filtering via first-order multitarget moments, IEEE Trans. on Aerosp. and Electron. Syst. 39 (4) (2003) 1152–1178.

[21] E. Maggio, E. Piccardo, C. Regazzoni, A. Cavallaro, Particle PHD filtering for multi-target visual tracking, in: Proc. Int. Conf. on Acoust., Speech, and Signal Process., Vol. 1, 2007, pp. I–1101–I–1104.

[22] M. Yang, F. Lv, W. Xu, Y. Gong, Detection driven adaptive multi-cue integration for multiple human tracking, in: Proc. Int. Conf. Comput. Vis., 2009, pp. 1554–1561.

[23] J. Zhang, L. L. Presti, S. Sclaroff, Online multi-person tracking by tracker hierarchy, in: Proc. IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance, 2012, pp. 379–385.

[24] A. Schumann, M. Bäuml, R. Stiefelhagen, Person tracking-by-detection with efficient selection of part-detectors, in: IEEE Int. Conf. on Advanced Video and Signal-Based Surveillance, 2013, pp. 43–50.

[25] N. J. Gordon, D. J. Salmond, A. F. M. Smith, Novel approach to nonlinear and non-gaussian bayesian state estimation, IEE Proc. F (Radar and Signal Process.) 140 (2) (1993) 107–113.

[26] Y. Jinxia, T. Yongli, X. Jingmin, Z. Qian, Research on particle filter based on an improved hybrid proposal distribution with adaptive parameter optimization, in: Proc. Int. Conf. on Intell. Computation Technol. and Automation, 2012, pp. 406–409.

[27] Y. Rui, Y. Chen, Better proposal distributions: Object tracking using unscented particle filter, in: Proc. Comput. Vis. Pattern Recognit., 2001, pp. 786–793.

[28] Y. Huang, P. Djuric, A hybrid importance function for particle filtering, IEEE Signal Process. Letters 11 (3) (2004) 404–406.

[29] S. Santhoshkumar, S. Karthikeyan, B. S. Manjunath, Robust multiple object tracking by detection with interacting markov chain Monte Carlo, in: Proc. Int. Conf. on Image Process., 2013.

[30] H. Han, Y.-S. Ding, K.-R. Hao, X. Liang, An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking, Comput. and Mathematics with Applicat. 62 (7) (2011) 2685–2695.

[31] S. Salti, A. Cavallaro, L. D. Stefano, Adaptive appearance modeling for video tracking: Survey and evaluation, IEEE Trans. on Image Process. 21 (10) (2012) 4334–4348.

[32] Y. Wu, J. Lim, M.-H. Yang, Online object tracking: A benchmark, in: Proc. Comput. Vis. Pattern Recognit., 2013, pp. 2411–2418.

[33] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.

[34] W. Zhong, H. Lu, M.-H. Yang, Robust object tracking via sparsity-based collaborative model, in: Proc. Comput. Vis. Pattern Recognit., 2012, pp. 1838–1845.

[35] J. Yang, D. Zhang, A. F. Frangi, J.-Y. Yang, Two dimensional PCA: A new approach to appearance-based face representation and recognition, IEEE Trans. Pattern Anal. Mach. Intell. 26 (1) (2004) 131–137.

[36] P. Dollár, S. Belongie, P. Perona, The fastest pedestrian detector in the west, in: Proc. British Machine Vis. Conf., 2010.

[37] M. A. Naiel, M. O. Ahmad, M. N. S. Swamy, Vehicle detection using TD2DHOG features, in: Proc. New Circuits and Syst. Conf., 2014, pp. 389–392.

[38] P. Dollár, R. Appel, S. Belongie, P. Perona, Fast feature pyramids for object detection, IEEE Trans. Pattern Anal. Mach. Intell. 36 (8) (2014) 1532–1545.

[39] T. Wang, I. Gu, P. Shi, Object tracking using incremental 2D-PCA learning and ML estimation, in: Proc. Int. Conf. on Acoust., Speech, and Signal Process., Vol. 1, 2007, pp. 933–936.

[40] C.-H. Kuo, C. Huang, R. Nevatia, Multi-target tracking by on-line learned discriminative appearance models, in: Proc. Comput. Vis. Pattern Recognit., 2010, pp. 685–692.

[41] M. Everingham, L. V. Gool, C. K. Williams, J.Winn, A. Zisserman, The Pascal visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.

[42] J. Ferryman, in: Proc. IEEE Workshop Performance Evaluation of Tracking and Surveillance, 2009.

[43] B. Benfold, I. Reid, Stable multi-target tracking in real-time surveillance video, in: Proc. Comput. Vis. Pattern Recognit., 2011, pp. 3457–3464.

[44] S. Sivaraman, M. Trivedi, A general active-learning framework for on-road vehicle recognition and tracking, IEEE Trans. on Intell. Transp. Syst. 11 (2) (2010) 267–276.

[45] B. Yang, R. Nevatia, Multi-target tracking by online learning of non-linear motion patterns and robust appearance models, in: Proc. Comput. Vis. Pattern Recognit., 2012, pp. 1918–1925.

[46] http://iris.usc.edu/people/yangbo/downloads.html, last retrieved March 14, 2016.

[47] http://research.milanton.de/data.html, last retrieved March 14, 2016.

[48] http://crcv.ucf.edu/data/ParkingLOT/index.php, last retrieved March 14, 2016.

[49] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Proc. Comput. Vis. Pattern Recognit., 2005, pp. 886–893.

[50] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the CLEAR MOT metrics, J. on Image and Video Process. (2008) 1–10.

[51] P. Dollár, Piotr's Image and Video Matlab Toolbox (PMT), http://vision.ucsd.edu/~pdollar/toolbox/doc/index.html, last retrieved, January 18, 2015.

[52] D. Zhang, Z.-H. Zhou, $(2D)^2$PCA: Two-directional two-dimensional PCA for efficient face representation and recognition, Neurocomputing 69 (1-3) (2005) 224–231.

[53] J. H. Yoon, M. H. Yang, J. Lim, K. J. Yoon, Bayesian multi-object tracking using motion context from multiple objects, in: Proc. IEEE Winter Conf. on Applicat. of Comput. Vis., 2015, pp. 33–40.

[54] S. H. Bae, K. J. Yoon, Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning, in: Proc. Comput. Vis. Pattern Recognit., 2014, pp. 1218–1225.

[55] X. Zhou, Y. Li, B. He, T. Bai, GM-PHD-based multi-target visual tracking using entropy distribution and game theory, IEEE Trans. on Ind. Informat. 10 (2) (2014) 1064–1076.

[56] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. V. Gool, Online multiperson tracking-by-detection from a single, uncalibrated camera, IEEE Trans. Pattern Anal. Mach. Intell. 33 (9) (2011) 1820–1833.

[57] D. G. Gomez, F. Lerasle, A. M. L. Peña, State-driven particle filter for multi-person tracking, in: Proc. Advanced Concepts for Intell. Vis. Syst., Vol. 7517 of Lecture Notes in Comput. Sci., 2012, pp. 467–478.

[58] F. Poiesi, R. Mazzon, A. Cavallaro, Multi-target tracking on confidence maps: An application to people tracking, Comput. Vis. and Image Under. 117 (10) (2013) 1257 – 1272.