A Multiple Motion Model Tracker Handling Occlusion and Rapid Motion Variation

Muhammad H. Khan
http://www.cs.nott.ac.uk/~mhk
Michel F. Valstar
http://www.cs.nott.ac.uk/~mfv
Tony P. Pridmore
http://www.cs.nott.ac.uk/~tpp

School of Computer Science
University of Nottingham
Nottingham, UK.

Abstract

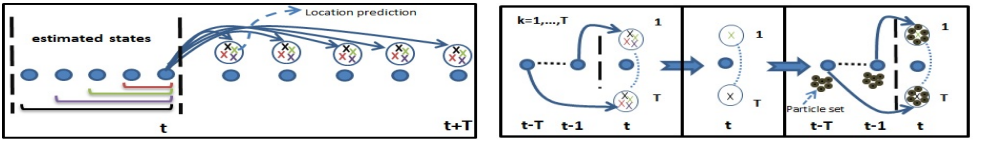
We propose a new tracking method capable of handling occlusions and non-constant target motion. This is achieved using multiple simple motion models, learned at different temporal scales and combined to capture possibly complex motion patterns. These motion models are learned online in a computationally inexpensive manner. Reliable recovery of tracking after occlusions is achieved by extending the bootstrap particle filter to propagate particles at multiple temporal scales, guided by the simple models. In complex environments targets can display changes in direction or speed unaccounted for by standard polynomial motion models. To demonstrate the generality of our framework and accommodate these changes, the proposed method is also applied to a more flexible, two-stage motion model. Extensive experiments have been carried out on both publicly available benchmarks and new video sequences. Results reveal that the proposed method successfully handles occlusions and a variety of rapid changes in target motion.

1 Introduction

Visual tracking is one of the most important unsolved problem in computer vision. Though it has received much attention, no framework has emerged which can robustly track across a broad spectrum of real world settings. Two major challenges for a visual tracker are variations in target motion and occlusions. In some applications, *e.g.* video surveillance and sports analysis, a target may undergo a variety of motions and be occluded at the same time.

While many solutions to the occlusion problem have been proposed, it remains unsolved. Some methods [17, 32] propose an explicit occlusion detection and handling mechanism. Reliable detection of occlusion is difficult in practice, and often produces false alarms. Some methods, based on adaptive appearance models [15, 28], use statistical reasoning to handle occlusions indirectly, by learning how appearance changes over time. Occlusions can, however, contaminate the appearance models, as these methods use blind update strategies.

Rapidly varying motion can be addressed using a single motion model with a large process noise. This approach requires large numbers of particles and is sensitive to background clutter. Alternative approaches include efficient proposals [23], or hybrid techniques with



(a) Multiple motion models are learned at multiple model-scales from the recent history of estimated states at time t, and are used to predict possible target locations at T prediction-scales.

(b) At time t, there are T different sets of predictions (left). One motion model is selected from each set belonging to $t-k$ at time t (middle), and used to propagate particles from $t-k$ to time t (right). A total of T motion models therefore generate particles at time t.

Figure 1: Graphical illustration of the proposed method.

hill climbing methods [22] to allocate particles near to the modes of the posterior. These approaches can, however, be computationally expensive.

We propose a new tracking method handling occlusion and non-constant motion. We believe that the most reliable way to recover from occlusion is to employ a flexible prediction method which estimates target location at temporal scales similar to the length of likely occlusions. To achieve this, motion models are learnt at multiple model-scales and used to predict possible target locations at multiple prediction-scales. The model-scale is the length of the sequence of recently estimated target states over which the motion model is learnt. The prediction-scale is the temporal distance, measured in frames of the input image sequence, over which a prediction is made. Reliable recovery of tracking after occlusions is achieved by extending the bootstrap particle filter to propagate particles to multiple prediction-scales, using models learnt at multiple model-scales. Fig. 1 summarises the approach.

The proposed framework places no restriction on the individual motion models used, and is shown here applying both polynomial motion models and the two-stage model of [13]. The two-stage model is more robust to acceleration, deceleration, and rapid changes in the direction of motion [13]. Our framework makes several contributions. Inspired by the success of multiple appearance model methods [14, 15], we introduce the concept of multiple motion models learned at different model-scales. These capture possibly complex motion patterns, and predict target location over multiple prediction-scales. Occlusion is handled implicitly, without using strong appearance models or an explicit occlusion detection mechanism.

2 Related Work

Occlusion handling may be direct or indirect. Indirect approaches can be divided into two categories. The first is based on adaptive appearance models which use statistical analysis [9, 15, 28] to reason about occlusion. The appearance models can, however, become corrupt during longer occlusions due to the lack of an intelligent update mechanism. Approaches in the second category divide the target into patches and either use a voting scheme [1] or robust fusion mechanism [10] to produce a tracking result. These can, however, fail when the number of occluded patches increases.

Some approaches address the occlusion of specific target types. Lim *et al.* [19] propose a human tracking system based on learning dynamic appearance and motion models. A three-dimensional geometric hand model was proposed by Sudderth *et al.* [30] to reason about occlusion in a non-parametric belief propagation tracking framework. Other researchers [6, 7] attempt to overcome occlusion using multiple cameras. As most videos are shot with a

single camera, and multiple cameras bring additional costs; this is not a generally applicable solution. Grabner *et al*. [8] and Yang *et al*. [31] employ spatio-temporal context to tackle the occlusion problem, but both methods rely on the tracking of auxiliary objects.

The explicit identification of occlusions requires a robust occlusion detection method. Collins *et al*. [32] presented a combination of local and global mode seeking techniques. Occlusion detection was achieved with a naive threshold based on the value of the objective function used in local mode seeking. Lerdsudwichai *et al*. [17] detected occlusions by using an occlusion grid with a drop in similarity value. This approach can produce false alarms because the required drop in similarity could occur due to natural appearance variation.

When target motion is difficult to model, a common solution is to use a single motion model with a large process noise. Examples of such models are random-walk (RW) [22, 26] and nearly constant velocity (NCV) [27, 29]. Increased process noise demands larger numbers of particles to maintain accurate tracking, which increases computational expense.

One approach to the increased variance in estimation caused by high process noise is to make an efficient and informed proposal distribution. Okuma *et al*. [23] designed a proposal distribution that mixed hypotheses generated by an AdaBoost detector and a standard autoregressive motion model to guide a particle filter based tracker. Kristan *et al*. [13] formulated a two-stage dynamic model to improve the accuracy and efficiency of bootstrap particle filters. The method fails when the target exhibits frequent spells of non-constant motion.

Several attempts have been made to learn motion models offline. Isard and Blake [11] use a hardcoded finite state machine (FSM) to manage transitions between a small set of learned models. Madrigal *et al*. [20] guide a particle filter based target tracker with a motion model learned offline. Pavlovic *et al*. [24] switch between motion models learned from motion capture data. Their approach is application specific, in that it learns only human motion. An obvious limitation of offline learning is that models can only be used to track the specific class of targets for which they are trained.

When tracking, knowledge of target motion can reduce the search space. To capture the ways a target can move, an interacting multiple model (IMM) approach based on Kalman [18] and particle filters [21] has been proposed. Here, each tracker employs a different motion model, and results are combined based on their performance. The particle filter IMM is not computationally feasible due to the calculation of large numbers of likelihood functions.

Our approach differs from previous work in using the recent history of the target to learn multiple simple motion models, whose predictions are pooled over multiple temporal scales to define the search space of a single particle filter. It is thus an online learning approach not restricted to any specific target class. A novel selection criterion determines the motion model adopted at each time point, without need for a hardcoded FSM.

3 Problem Formulation

Our aim is to find the best state of the target at time t given observations up to t. State at time t is given by $\mathbf{X}_t = \{X_t^x, X_t^y, X_t^s\}$,where X_t^x,X_t^y, and X_t^s represent the x,y location and scale of the target, respectively. The posterior probability $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ given the state $\mathbf{X}_t$ at time t, and observations $\mathbf{Y}_{1:t}$ up to t, is estimated using the Bayesian formulation

$$p(\mathbf{X}_t|\mathbf{Y}_{1:t}) \propto p(\mathbf{Y}_t|\mathbf{X}_t) \int p(\mathbf{X}_t|\mathbf{X}_{t-1}) p(\mathbf{X}_{1:t-1}|\mathbf{Y}_{1:t-1}) \mathrm{d}X_{t-1}, \tag{1}$$

where $p(\mathbf{Y}_t|\mathbf{X}_t)$ denotes the observation model and $p(\mathbf{X}_t|\mathbf{X}_{t-1})$ is a motion model. The best state of the target $\hat{\mathbf{X}}_t$ is obtained using Maximum a Posteriori (MAP) estimate over the N_t weighted particles which approximate $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$,

$$\hat{\mathbf{X}}_t = \arg\max_{\mathbf{X}_t^{(i)}} p(\mathbf{X}_t^{(i)}|\mathbf{Y}_{1:t}) \; for i = 1,...,N_t, \tag{2}$$

where $\mathbf{X}_t^{(i)}$ is the i_{th} particle. An accurate value of this MAP estimate depends on an accurate estimation of posterior probability in eq.(1). This becomes difficult when the target is occluded or exhibits dynamics which are not covered by commonly used motion models such as the RW or the NCV model.

4 Proposed Method

4.1 A Multiple Motion Model Framework

To reliably recover the target after occlusion, we introduce the concept of motion models learnt at a range of model-scales, and contribute a simple but powerful extension of the bootstrap particle filter. The proposed concept and extension are general, and applicable to any particle filter. The core idea is to combine sufficient particle sets at each time-point that at least one set will be valid, and allow recovery from occlusion. A valid particle set represents an accurate estimation of the posterior probability from some previous time-point, predicted by a motion model generated over an appropriate model-scale and unaffected by occlusion.

4.1.1 Learning Simple Motion Models

A simple motion model is characterized by a polynomial function of order d. It is learned at a given model-scale separately on the x-location and y-location of the target's state. This learning also considers how well each state is estimated in a given sequence and how far it is from the most recently estimated state [13]. For instance, a model of order 1, learned at model-scale m, predicts target x-location at time t thus

$$\tilde{x_t} = \beta_o^m + \beta_1^m t + \mathcal{N}(0, \sigma_m^2), \tag{3}$$

where β_1 is the slope, β_o the intercept, and $\mathcal{N}(0, \sigma_m^2)$ is a zero-mean Gaussian distribution with variance σ_m^2. Model parameters can be learnt inexpensively via weighted least squares.

4.1.2 Model Set Reduction

A set of learned motion models at time t is represented by $\mathbf{M}_t^{j=1,...,|\mathbf{M}_t|}$, where $|.|$ is the cardinality of the set. If the cardinality is G, and each model predicts target location $l(\tilde{x}, \tilde{y})$ at T prediction-scales, then at any given time-step there will be $G \times T$ predictions. These are not all equally accurate. Thus, to reduce the chance of false positive predictions, the most suitable motion model $\mathbf{R}_t^k$ is selected from a set belonging to the k_{th} previous time-step according to the criterion:

$$\mathbf{R}_t^k = \arg\max_{l_t^{j,k}} p(\mathbf{Y}_t|l_t^{j,k}) \; j = 1,...G \text{ and } k = 1,...T \tag{4}$$

where j,k denotes the j_{th} motion model from k_{th} previous time-step, and $p(\mathbf{Y}_t|l_t^{j,k})$ measures the visual likelihood that the target is located at the predicted location $l_t^{j,k}$.

4.1.3 Propagation of particles

In the bootstrap particle filter [5], the posterior probability at time t is estimated by a set of particles $\mathbf{X}_t^{(i)}$ and their weights $\omega_t^{(i)}$,$\{\mathbf{X}_t^{(i)}, \omega_t^{(i)}\}_{i=1}^N$, such that all the weights in the particle set sum to one. At time $t+1$, the particles are resampled to form an unweighted representation of the posterior $\{\mathbf{X}_t^{(i)}, 1/N\}_{i=1}^N$. They are then propagated using the motion model $p(\mathbf{X}_{t+1}|\mathbf{X}_t)$ to approximate a prior distribution $p(\mathbf{X}_{t+1}|\mathbf{Y}_t)$. Finally, they are weighted according to the observation model $p(\mathbf{Y}_{t+1}|\mathbf{X}_{t+1})$, approximating the posterior probability at $t+1$.

Here the particle set $\{\mathbf{X}_t^{(i)}, \omega_t^{(i)}\}_{i=1}^N$ at t is propagated not only to $t+1$ but to the next T time-steps. At a given time t, the selected motion model $\mathbf{R}_t^k$ belonging to the k_{th} previous time-step will propagate particles belonging to the k_{th} previous time-step as follows[1]:

$$X_{t,k}^x = X_{t-k}^x + g(\mathbf{R}_t^k)k + \mathcal{N}(0, \sigma_x^2 k), \tag{5}$$

where X^x is the horizontal part of the target state, $g()$ indicates the slope of the model, and $\mathcal{N}(0,\sigma_x)$ is a Gaussian distribution with zero-mean and σ_x^2 variance.

Propagation from the last T time-steps, generates T particle sets at time t. All particles are weighted using the observation model $p(\mathbf{Y}_t|\mathbf{X}_t)$ to approximate the posterior probability $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$. If the target was occluded for less than or equal to $T-1$ frames, it may be recovered by a set of particles unaffected by the occlusion. The proposed framework is summarised in Algorithm 1.

Algorithm 1 Multiple Motion Model tracker

Input: The resampled sets of particles after estimation of posterior from T previous time-steps $\{\mathbf{X}_{t-k}^{(i)}, \frac{1}{N}\}_{i=1}^N$ where $k = 1, ..., T$.

Output: Best state $\hat{\mathbf{X}}_t$ at time t.

for $k = 1$ *to* T

for $j = 1$ *to* G

- Measure visual likelihood $p(\mathbf{Y}_t|l_t^{j,k})$, where $l_t^{j,k}$ denotes the predicted location at time t by j_{th} motion model from k_{th} previous time-step.

end

- Select the most suitable motion model $\mathbf{R}_t^k$ at time t using eq.(4).
- Propagate the particle set from k_{th} previous time-step $\{\mathbf{X}_{t-k}^{(i)}, \frac{1}{N}\}_{i=1}^N$ using eq.(5) by taking the slope of selected motion model $\mathbf{R}_t^k$ to time t.

end

- Assign weights to all the particles to approximate the posterior $\{\mathbf{X}_t^{(i)}, \omega_t^{(i)}\}_{i=1}^{N\times T}$.
- Calculate the best state $\hat{\mathbf{X}}_t$ using eq.(2).
- Retain first N particles after the resampling step.
- Learn simple motion models using the recent history of estimated states.

4.2 Improving the Simple Motion Model

Targets can exhibit motions - accelerations, decelerations, and rapid changes in direction - which are not well-represented by the widely used polynomial motion models. We therefore

[1]To demonstrate the basic idea of the proposed method, eq.(5) is used for X^x and X^y part of the target state, and for X^s a simple RW model with variance $\sigma_s^2 k$ has been used. Though the method can be easily extended to learn simple motion models for recently estimated scales, and eq.(5) can be used to propagate X^s part of the target state.

also apply our multiple motion model framework to a more flexible, two-stage motion model [13]. The state of the target now includes an additional internal velocity term v in both the x and y directions. The flexible motion model with state $\mathbf{X}_t$ can be written as [13]:

$$\mathbf{X}_t = \Phi\mathbf{X}_{t-1} + \Gamma\hat{v}_{t-1} + W_t,$$
$$\Phi = \begin{bmatrix} 1 & \frac{1-e^{-\Delta t\beta}}{\beta} \\ 0 & e^{-\Delta t\beta} \end{bmatrix}, \Gamma = \begin{bmatrix} \frac{\Delta t\beta - 1 + e^{-\Delta t\beta}}{\beta} \\ e^{-\Delta t\beta} \end{bmatrix}. \tag{6}$$

where Φ is the state-transition matrix, Γ is the discrete time gain through which the rigid velocity $\hat{v}_{t-1}$ enters the system, Δt is the time-step length, and W_t is the white noise sequence. For details, see [13], page 4. The correlation time parameter β can be adjusted to tune the properties of the two-stage model.

With the inclusion of the two-stage motion model in the proposed method, we have two ways to estimate target location: the rigid prediction $\check{l}_t = (\check{x}_t, \check{y}_t)$ and the flexible estimate $\hat{l}_t = (\hat{X}_t^x, \hat{X}_t^y)$. From the selected polynomial motion models $\mathbf{R}_t^{k=1:T}$ available at time t, the prediction of the model with the highest visual likelihood score is taken as the rigid prediction $\check{l}_t = (\check{x}_t, \check{y}_t)$. The flexible motion model propagates particles from the k_{th} previous time-step to time t by taking the slope of the selected polynomial motion model as the rigid velocity. After propagation from T previous time-steps, the flexible estimate of the target state $\hat{\mathbf{X}}_t = \{\hat{X}_t^x, \hat{X}_t^y, \hat{X}_t^s\}$ is computed using eq.(2). Now the normalized location of the target $\hat{n}_t$ is calculated by reducing the variance of the flexible estimate of the target location $\hat{l}_t = (\hat{X}_t^x, \hat{X}_t^y)$ by fusing it with the rigid prediction $\check{l}_t$ of the target location:

$$\hat{n}_t = \frac{\check{l}_t \psi_{\check{l}_t} + \hat{l}_t \psi_{\hat{l}_t}}{\psi_{\check{l}_t} + \psi_{\hat{l}_t}}, \tag{7}$$

where $\psi_{\check{l}_t}$ is the visual likelihood score that the target is located at $\check{l}_t$, and $\psi_{\hat{l}_t}$ is the visual likelihood score that the target is located at $\hat{l}_t$.

5 Experimental Results

The proposed method addresses the occlusion problem using motion information only. The appearance model used in all experiments was, therefore, the colour histogram used in [25]. Model-scales ranged from 2 to 5, the simple motion model was linear. Four simple motion models were used. The β parameter of the two-stage model was fixed at 10, giving high weight to the rigid velocity $\hat{v}$, estimated by the simple motion model, and very low weight to the internal velocity v. As a result, it becomes strongly biased towards the predicted location, but still allows some deviation. We compared the proposed method to three baseline and five state-of-the-art trackers. The first two baseline trackers, T_{RW} and T_{NCV}, were colour based particle filters from [25], but use different motion models. The first tracker T_{RW} used a random-walk model while the second tracker T_{NCV} used a nearly constant velocity model. The third baseline tracker T_{TS} was the two-stage dynamic model proposed by [13]. The parameter K in [13] was set to 5. The state-of-the-art trackers are L1-APG(L1 tracker using Accelerated Proximal Gradient Approach) [5]),VTD(Visual Tracking Decompostion [15]), IVT(Incremental Subspace Visual Tracker [28]), FragT(Fragment-based Tracker [1]), and SemiBoost(Semisupervised boosting Tracker [9]). The samples used for L1-APG, VTD, and IVT were 640. The parameters of the competing methods were adjusted to produce

(a) TU-Cro♯25 (b) TU-Cro♯48 (c) TU-Cro♯108 (d) car♯168 (e) car♯191 (f) car♯231

Figure 2: Tracking through multiple partial occlusions. Groundtruth(purple), T_{MM2}(cyan), FragT(white), SemiBoost(yellow), L1-APG(blue), VTD(red) and IVT(green).

the best tracking performance. The supplementary material contains list of parameters of different trackers used.

Nine video sequences were used. Six are publicly available (*TUD-Campus* [2], *TUD-Crossing* [2], *PETS 2001 Dataset 1*[2], *Person* [16], *car* [12], and *PETS 2009 Dataset S2*[3]) and three are our own (*squash*, *ball1*, and *ball2*). All involve frequent short and long term occlusions (partial and full) and/or variations in target motion.

5.1 Quantitative and Qualitative Evaluation

Table 1 summarises tracking accuracy achieved over the 7 sequences. T_{MMP} denotes the proposed method applied over a first-order polynomial motion model, and T_{MM2} the use of the two-stage model. T_{MMP} outperformed competing methods in most sequences, because it efficiently allocated particles to overcome occlusions. VTD, and IVT performed badly because inappropriate appearance model updates during longer occlusions causes drift from which they cannot recover. Although SemiBoost uses explicit re-detection once the target is lost, its accuracy was low due to false positive detections. FragT and L1-APG produced the lowest error in the *car*, and *TUD-Crossing* sequences, respectively, which involved partial occlusions (Fig. 2). FragT uses a patch based target representation, and L1-APG employs a robust minimization model influenced by an explicit occlusion detection mechanism. In contrast, T_{MMP} and T_{MM2} use a very simple, generic appearance model, and no explicit occlusion handling mechanism.

Table 1: **Tracking accuracy in the presence of occlusion**. Mean centre location error in pixels is given, averaged over all frames of all videos showing occlusions. Each tracker was run five times and the results were averaged. The best results are marked in bold. T denotes the prediction-scales, and N is the number of particles propagated from $t-k$ to t in our proposed method. N is fixed at 20, and N_t is the total number of particles accumulated at time t in our proposed method. The number of particles used in baseline trackers was equal to N_t.

Sequence	T_{NCV}	T_{RW}	T_{TS}	*IVT*	*L1-APG*	*VTD*	*Semi*	*FragT*	T_{MMP}	T_{MM2}	T	$N_t = N \times T$
ball2	91	71	125	104	71	66	78	106	30	**27**	32	640
TUD-Camp	141	119	31	186	100	186	61	112	19	**18**	8	160
TUD-Cross	43	75	106	41	**2**	63	62	5	30	28	25	500
PETS 2001	43	131	112	76	60	83	114	67	**20**	23	32	640
Person	90	33	95	83	103	85	177	84	9	**8**	20	400
PETS 2009	75	37	56	80	81	94	29	10	8	**7**	14	280
car	37	43	87	81	31	47	38	**16**	26	25	20	400

Tracking accuracy was also measured when the target was occluded and underwent motion variation at the same time (Table 2). T_{MMP} produced higher accuracy than the other

[2] *PETS 2001 Dataset 1* is available from http://ftp.pets.rdg.ac.uk/

[3] *PETS 2009 Dataset S2* is available from http://www.cvg.rdg.ac.uk/PETS2009/

Table 2: **Accuracy through simultaneous motion variation and occlusion**. Mean centre location error (pixels) is given. Each tracker was run five times and the results were averaged.

Sequence	T_{NCV}	T_{RW}	T_{TS}	*IVT*	*L1-APG*	*VTD*	*Semi*	*FragT*	T_{MMP}	T_{MM2}	T	$N_t = N \times T$
squash	27	52	41	122	60	21	68	35	14	**12**	5	100
ball1	74	87	98	211	124	69	67	210	17	**17**	14	280

methods. By propagating particles over multiple prediction-scales, T_{MMP} efficiently allocates particles to reduce the search space while covering a wide range of target dynamics. VTD performed well in *squash* sequence because it combines two motion models of different variances to form multiple basic trackers which search a large state space efficiently. SemiBoost produced second best accuracy in *ball1*, as appearance varies little throughout the sequence.

Note that T_{MM2} performs only slightly better than T_{MMP}. In general, T_{MMP} produces an accurate approximation of the likely target path. When the target deviates considerably from its predicted location, T_{MM2} is a little more accurate; the flexible motion model spreads the particles more widely to compensate. For instance in Fig. 3, when the player suddenly changes direction, T_{MM2} demonstrates greater tracking accuracy.

(a) # 16 (b) # 16 (c) # 17 (d) # 17 (e) # 18 (f) # 18

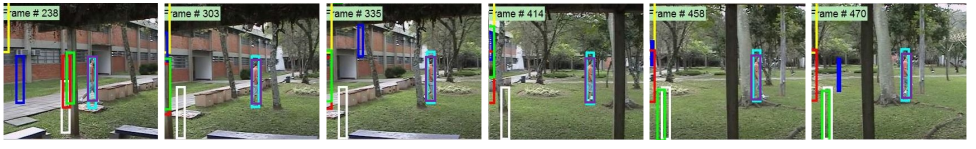
Figure 3: T_{MM2}(cyan) is more accurate than T_{MMP}(white) when faced with sudden changes in direction. T_{MMP} produces concentrated sets of particles while the flexible motion model in T_{MM2} provides more spread.

Tracking is particularly difficult when the time between consecutive occlusions is small. In *TUD-Campus*, the tracked person suffers two occlusions only 17 frames apart (Fig. 4a). VTD, L1-APG, and IVT failed due to incorrect appearance model updates. FragT drifted when the target was completely occluded, and SemiBoost could not re-locate the target reliably in the surrounding clutter; it relies completely on the detector once the target is lost. T_{MM2} recovers the target after each occlusion. Video surveillance data often requires tracking through partial and/or full occlusions. In the *PETS 2001 Dataset 1* sequence (Fig. 4b) the target (car) first stays partially occluded for a considerable time, and is then completely occluded by a tree. T_{MM2} successfully re-acquires the target. Occlusions of varying lengths are common in real-world tracking scenarios. In the *person* sequence, a person moves behind several trees and is shot with a moving camera. As shown in Fig. 5, competing methods lose the target after first occlusion (Frame # 238), while T_{MM2} shows robustness in coping with varying lengths of occlusions. Fig. 6 gives cumulative position errors (over time) obtained from T_{MM2}, VTD, L1-APG, IVT, FragT, and SemiBoost.

The ability of T_{MM2} to cope with simultaneous occlusion and non-constant target motion was tested by making two challenging sequences: *squash* and *ball1*. In these sequences, the target accelerates, decelerates, changes direction suddenly, and is completely occluded multiple times. Fig. 7 illustrates tracking results. T_{MM2} provided more accurate tracking than the other methods on both sequences. This is because the efficient allocation of particles at

(a) TUD-Campus♯12♯29♯45 (b) PETS'01♯36♯91♯178

Figure 4: Tracking results in a crowded (a) and a surveillance environment (b). Groundtruth(purple), T_{MM2}(cyan), FragT(white), SemiBoost(yellow), L1-APG(blue), VTD(red) and IVT(green).



(a) # 238 (b) # 303 (c) # 335 (d) # 414 (e) # 458 (f) # 470

Figure 5: Tracking results with occlusions of different lengths. Groundtruth(purple), T_{MM2}(cyan), FragT(white), SemiBoost(yellow), L1-APG(blue), VTD(red) and IVT(green).

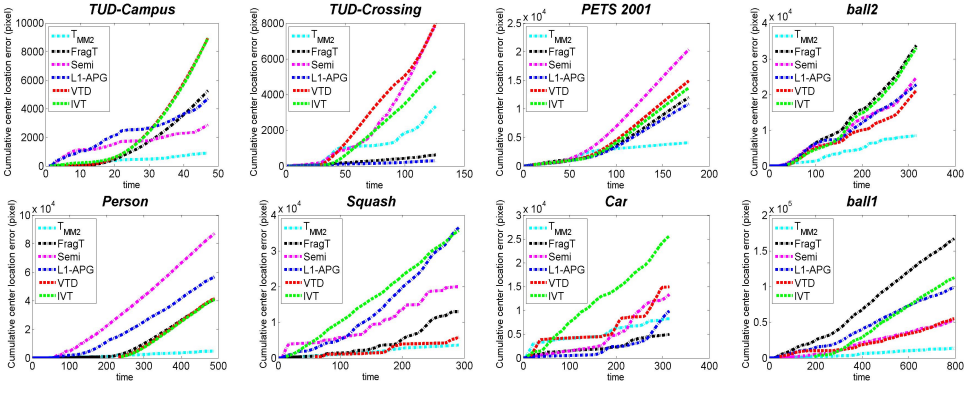
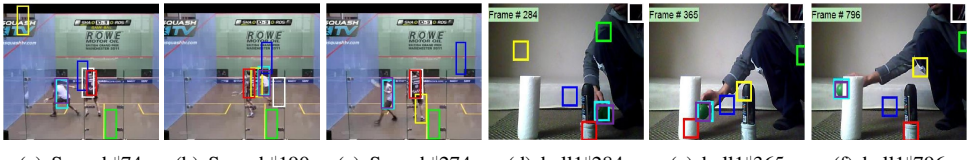


Figure 6: Cumulative errors associated with FragT, SemiBoost, VTD, L1-APG, IVT, and T_{MM2}.



(a) Squash♯74 (b) Squash♯190 (c) Squash♯274 (d) ball1♯284 (e) ball1♯365 (f) ball1♯796

Figure 7: Tracking results in case of motion variations and frequent occlusions. Groundtruth(purple), T_{MM2}(cyan), FragT(white), SemiBoost(yellow), L1-APG(blue), VTD(red) and IVT(green).

multiple prediction-scales allows covering a wider range of target motion.

Experimental results show the robust performance of the proposed framework during occlusions. However, the proposed method can fail when faced with very long duration occlusions. In addition, it can distract to a visually similar object after occlusion, if the state estimations during the period of occlusion are poor.

6 Conclusion

We propose a tracking framework that combines motion models learned over multiple model-scales and applied over multiple prediction-scales to handle occlusion and variation in target motion. The core idea is to combine sufficient particle sets at each time-point that at least one set will be valid, and allow recovery from occlusion and/or motion variation. These particle sets are not, however, simply spread widely across the image: each represents an estimation of the posterior probability from some previous time-point, predicted by a motion model generated over an appropriate model-scale.

The framework can be applied to any motion and appearance model pair. Simple, fixed appearance models have been used here for generality, and polynomial and two-stage motion models have been employed to demonstrate the flexibility of the approach. The proposed method has shown superior performance over state-of-the-art trackers in challenging tracking environments. That there is little difference between results obtained using polynomial and two-stage motion models suggests that this high level of performance is due to the framework, rather than its components.

References

[1] A. Adam, E. Rivlin, and I. Shimshoni. Robust fragments-based tracking using the integral histogram. In *CVPR* 2006.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *CVPR*, 2008.

[3] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. Signal Proc.*, 50(2): 174–188, 2002.

[4] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Visual tracking with online multiple instance learning. In *CVPR*, 2009.

[5] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1830–1837. IEEE, 2012.

[6] S. L. Dockstader and A. M. Tekalp. Multiple camera tracking of interacting and occluded human motion. *Proceedings of IEEE*, 89(10):1441–1455, 2001.

[7] F. Fleuret, R. Lengagne J. Berclaz, and P. Fua. Multicamera people tracking with a probabilistic occupancy map. *IEEE Trans. PAMI*, 30(2):267–282, 2010.

[8] H. Grabner, J. Matas, L. Van Gool, and P. Cattin. Tracking the invisible: Learning where the object might be, . In *CVPR*, 2010.

[9] Helmut Grabner, Christian Leistner, and Horst Bischof. Semi-supervised on-line boosting for robust tracking, . In *ECCV*, 2008.

[10] B. Han and L. S. Davis. Probabilistic fusion-based parameter estimation for visual tracking. CVIU, 113(1):435–445, 2009.

[11] M. Isard and A. Blake. A mixed-state condensation tracker with automatic model-switching. In *ICCV*, 1998.

[12] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Tracking-learning-detection. *IEEE Trans. PAMI*, 34(7):1409–1422, 2012.

[13] M. Kristan, S. Kovačič, A. Leonardis, and J. Perš. A two-stage dynamic model for visual tracking. *IEEE Trans. Systems, Man and Cybernetics, B*, 40(6):1505–1520, 2010.

[14] J. Kwon and K.M. Lee. Tracking by sampling trackers, . In *ICCV*, 2011.

[15] J. Kwon and K.M. Lee. Visual tracking decomposition, . In *CVPR* 2010.

[16] Leandro L., ClÃąudio Rosito, and C.Jose. Robust adaptive patch-based object tracking using weighted vector median filters. In *SIBGRAPI*, 2011.

[17] C. Lerdsudwichai, M. Abdel-Mottaleb, and A. Ansari. Tracking multiple people with recovery from partial and total occlusion. *Pattern Recognition*, 38(7):1059–1070, 2005.

[18] W. R. Li and Y. Bar-Shalom. Performance prediction of the interacting multiple model algorithm. *IEEE Trans. Aerospace and Electronic Systems*, 29(3):755–771, 1993.

[19] H. Lim, O. I. Camps, M. Sznaier, and V. I. Morariu. Dynamic appearance modeling for human tracking. In *CVPR* 2006.

[20] F. Madrigal, M. Rivera, and J. Hayet. Learning and regularizing motion models for enhancing particle filter-based target tracking. In *PSIVT*, 2011.

[21] S. McGinnity and G. Irwin. Multiple model bootstrap filter for maneuvering target tracking. *IEEE Trans. Aerospace and Electronic Systems*, 36(3):1006–1012, 2000.

[22] A. Naeem, T. Pridmore, and S. Mills. Managing particle spread via hybrid particle filter/kernel mean shift tracking. In *BMVC*, 2007.

[23] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, 2004.

[24] Vladimir Pavlovic, James M. Rehg, and John Maccormick. Learning switching linear models of human motion. In *NIPS*, 2000.

[25] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *ECCV*, 2002.

[26] P. Perez, J. Vermaak, and A. Blake. Data fusion for visual tracking with particles. *Proc. of the IEEE*, 92(3):495–513, 2004.

[27] F. Pernkopf. Tracking of multiple targets using online learning for reference model adaptation. *IEEE Trans. Systems, Man and Cybernetics, B*, 38(6):1465–1475, 2008.

[28] David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *International Journal of Computer Vision*, 77(1): 125–141, 2008.

[29] C. Shan, T. Tan, and Y. Wei. Real-time hand tracking using a mean shift embedded particle filter. *Patt. Recogn*, 40(7):1958–1970, 2007.

[30] E. Sudderth, M. Mandel, W. Freeman, and A. Willsky. Distributed occlusion reasoning for tracking with nonparametric belief propagation. In *NIPS*, MIT Press 2004.

[31] M. Yang, Y. Wu, and G. Hua. Context-aware visual tracking. *IEEE Trans. PAMI*, 31 (7):1195–1209, 2009.

[32] Zhaozheng Yin and Robert T. Collins. Object tracking and detection after occlusion via numerical hybrid local and global mode-seeking. In *CVPR*, 2008.