# Interval Tracker: Tracking by Interval Analysis

Junseok Kwon
Computer Vision Lab, ETH Zurich
kwonj@vision.ee.ethz.ch

Kyoung Mu Lee
Computer Vision Lab, Seoul National University
kyoungmu@snu.ac.kr

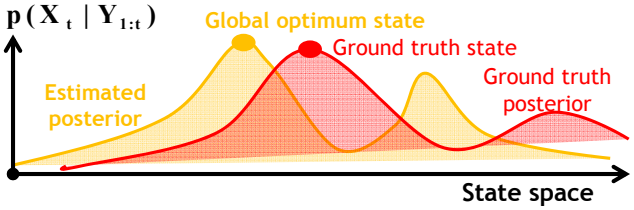
## Abstract

*This paper proposes a robust tracking method that uses interval analysis. Any single posterior model necessarily includes a modeling uncertainty (error), and thus, the posterior should be represented as an interval of probability. Then, the objective of visual tracking becomes to find the best state that maximizes the posterior and minimizes its interval simultaneously. By minimizing the interval of the posterior, our method can reduce the modeling uncertainty in the posterior. In this paper, the aforementioned objective is achieved by using the M4 estimation, which combines the Maximum a Posterior (MAP) estimation with Minimum Mean-Square Error (MMSE), Maximum Likelihood (ML), and Minimum Interval Length (MIL) estimations. In the M4 estimation, our method maximizes the posterior over the state obtained by the MMSE estimation. The method also minimizes interval of the posterior by reducing the gap between the lower and upper bounds of the posterior. The gap is reduced when the likelihood is maximized by the ML estimation and the interval length of the state is minimized by the MIL estimation. The experimental results demonstrate that M4 estimation can be easily integrated into conventional tracking methods and can greatly enhance their tracking accuracy. In several challenging datasets, our method outperforms state-of-the-art tracking methods.*

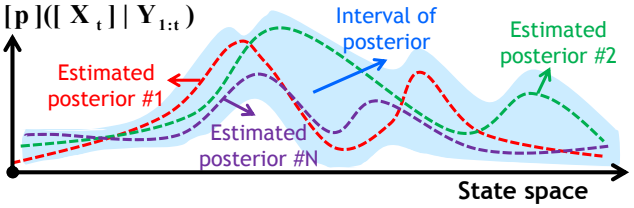
## 1. Introduction

Object tracking is one of the important problems in computer vision. Many researchers have recently addressed this problem by using real-world scenarios rather than performing laboratory simulations [4, 7, 9, 12, 28, 18, 23, 29, 32, 11].

To robustly track a target in a real-world scenario, most conventional tracking methods formulate the tracking problem by the Bayesian approach, where the goal is to find the best state that maximizes the posterior $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$. This approach is called the MAP estimation, that is, $\hat{\mathbf{X}}_t = \arg\max_{\mathbf{X}_t} p(\mathbf{X}_t|\mathbf{Y}_{1:t})$, where $\hat{\mathbf{X}}_t$ denotes the best state (MAP state) at time $t$ given the observation $\mathbf{Y}_{1:t}$. The posterior $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ is efficiently obtained by Bayesian filtering. Given the state at time $t$ and the observation up to time $t$, the Bayesian filter updates the posterior $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ with the following formula: $p(\mathbf{X}_t|\mathbf{Y}_{1:t}) \propto p(\mathbf{Y}_t|\mathbf{X}_t) \times \int p(\mathbf{X}_t|\mathbf{X}_{t-1})\, p(\mathbf{X}_{t-1}|\mathbf{Y}_{1:t-1})\, d\mathbf{X}_{t-1}$, where $p(\mathbf{Y}_t|\mathbf{X}_t)$, $p(\mathbf{X}_t|\mathbf{X}_{t-1})$, $\mathbf{X}_t$,and $\mathbf{Y}_t$ denote the appearance, motion, state, and observation models, respectively. Thus, the posterior is determined by the distributions associated with the appearance, motion, state, and observation models. Conventional tracking systems [33] typically assume that their employed distribution models are correct. However, this assumption is not valid in practice [3]. Note that any single posterior model can have a modeling error when distributions associated with the appearance, motion, state, and observation models are contaminated [3], and the estimated posterior may be incorrect, as illustrated in Fig.1(a).




(a) Conventional representation using a single posterior




(b) New representation using interval of a posterior

Figure 1. **Problem of conventional posterior representation.** (a) The estimated posterior necessarily has a modeling uncertainty. Hence, the global optimum state of the estimated posterior may not correspond to the global optimum state of a true posterior. (b) To deal with the modeling uncertainty, a posterior should be represented by the multiple candidates of posteriors. Finally, the infinite candidates of estimated posteriors form the lower and upper bounds of the posterior and become the interval of the posterior (blue region).

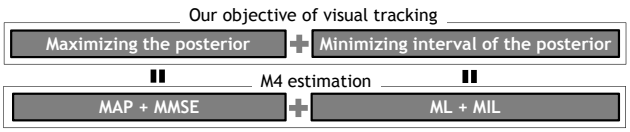



Figure 2. **Basic idea of the proposed tracker.** Our objective of visual tracking is to maximize the posterior and minimize the interval of the posterior simultaneously. To achieve this goal, the M4 estimation is proposed, which combines MMSE-MAP with ML-MIL. In M4 estimation, MMSE-MAP find the MMSE state that maximizes the posterior using the MAP estimation while ML-MIL find the state that minimizes the interval of the posterior. The interval of the posterior can be minimized by maximizing the likelihood using the ML estimation. The interval can be also reduced by minimizing the interval length of the state using the MIL estimation.

Hence, even though we can find the optimal MAP state of this incorrect posterior with recent advanced optimization techniques, the solution does not always correspond to the ground-truth state of a target.

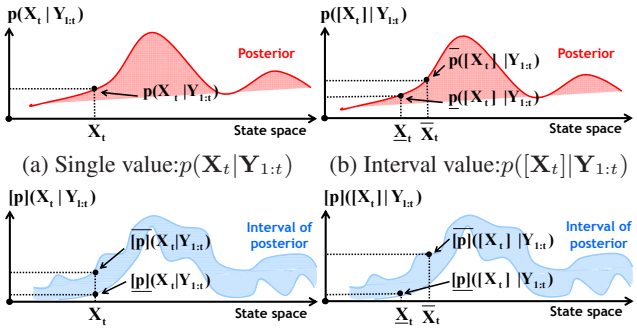
In the present study, we consider the modeling of the uncertainty in the appearance (likelihood) and state (prior) models to overcome the posterior modeling problem, and propose to use the interval of the posterior, as illustrated in Fig.1(b). The uncertainty in the appearance and state models occurs when information about the appearance and state of the target is initially insufficient or only partially available. For example, if the target is severely occluded in the initialization step, due to the inaccurate appearance model, the tracking methods can hardly determine the unique state and appearance of the target. The modeling uncertainty occurs also when information about the state and appearance of the target is corrupted during the tracking process. In this case, the resulting trackers cannot perfectly estimate and update the appearance and state models. On the other hand, our method can overcome the modeling uncertainty problem by representing the posterior as an interval. Our method cast the tracking problem into finding the best state that maximizes the posterior while minimizing the interval of the posterior. The best state can then be efficiently obtained by the proposed M4 estimation, which combines the MAP with ML, MMSE, and MIL, as illustrated in Fig.2.

The contribution of the proposed method is fourfold. First, the tracking problem is designed via the interval-based formulation. The posterior is defined using interval representation in (1). With the interval representation, our method can reduce the modeling error of the posterior and track the targets accurately. Second, the M4 estimation is proposed to find the best state that maximizes the posterior and minimizes its interval. In Section 4, we show that MMSE-MAP and ML-MIL find the state that maximizes the posterior and minimizes its interval, respectively. Third, the interval linearization technique [31] is applied to the tracking problem, which efficiently decomposes the interval based posterior into two terms: mean posterior without an interval and interval of the posterior. Mean posterior is similar to the conventional one. Interval of the posterior, however, is not considered by conventional tracking methods. Finally, our tracking method is highly applicable; it can be easily integrated into existing tracking algorithms and can greatly improve their tracking performance. The aforementioned advantages of our method are demonstrated through extensive experiments.

## 2. Related Work

**Tracking methods using Bayesian Model Averaging (BMA) [25]:** The basic idea of BMA is to consider multiple candidates of posteriors and to average them according to some criterion [3]. By averaging multiple candidates, the statistical error (uncertainty) decreases at the rate of the square root of the number of candidates. For example, if $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ is modeled by the weighted average of posteriors $\{p_i(\mathbf{X}_t|\mathbf{Y}_{1:t})\}_{i=1}^N$: $p(\mathbf{X}_t|\mathbf{Y}_{1:t}) = \sum_{i=1}^N w_i p_i(\mathbf{X}_t|\mathbf{Y}_{1:t})$, where $w_i$ is the weight of the $i$-th estimated posterior, then the statistical error of $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ decreases at the rate of $\sqrt{N}$. Following this approach, the VTD tracker in [14] averaged multiple appearance and motion models. Each appearance and motion model covers a specific appearance of the object and a different type of motion, respectively. The VTS tracker in [15] averaged multiple observation and state models as well, which make tracking methods less sensitive to noise and motion blur.

**Tracking methods using Interval Analysis (IA) [22]:** When different but reliable posteriors yield substantially different answers, it is better and more reasonable to consider all the possible candidates of posteriors instead of deciding one of the answers to be true [3]. For example, the posterior $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ can be estimated by an interval as $\underline{p}(\mathbf{X}_t|\mathbf{Y}_{1:t}) \leq p(\mathbf{X}_t|\mathbf{Y}_{1:t}) \leq \overline{p}(\mathbf{X}_t|\mathbf{Y}_{1:t})$, where $\underline{p}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ and $\overline{p}(\mathbf{X}_t|\mathbf{Y}_{1:t})$ are the lower and upper bounds of the estimated posterior, respectively. IA is different from BMA in the following aspect; IA considers an infinite number of candidates by interval representation, whereas BMA only utilizes a finite number of posterior candidates. Hence, IA can be regarded as a proper and powerful extension of BMA. However, efforts to utilize IA in the visual tracking problem have been few. The MUG tracker in [16] robustly tracked the target using the lower and upper bounds of the likelihood. Compared with the MUG, our method has two advantages. The first is to use the state interval as well, which is not considered in [16]. The second is to infer the posterior interval by integrating both likelihood and state intervals into the Bayesian formulation in (1). Although integrating both likelihood and state intervals into the posterior is not a trivial task, our method successfully infers the posterior interval within the interval analysis framework. By the help of these two advanced ideas, our method produces




(a) Single value:$p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ (b) Interval value:$p([\mathbf{X}_t]|\mathbf{Y}_{1:t})$

(c) Interval value:$[p](\mathbf{X}_t|\mathbf{Y}_{1:t})$ (d) Interval value:$[p]([\mathbf{X}_t]|\mathbf{Y}_{1:t})$

Figure 3. **Four different types of the posterior.** (a) $p(\mathbf{X}_t|\mathbf{Y}_{1:t})$ does not employ the modeling uncertainty. (b) $p([\mathbf{X}_t]|\mathbf{Y}_{1:t})$ only employs the modeling uncertainty of the *state*. (c) $[p](\mathbf{X}_t|\mathbf{Y}_{1:t})$ only employs the modeling uncertainty of the *posterior*. (d) $[p]([\mathbf{X}_t]|\mathbf{Y}_{1:t})$ employs the modeling uncertainty of both the *state* and the *posterior*.

more accurate results than the state-of-the-art methods including [16].

**Other tracking methods to solve ambiguities in visual tracking models:** IVT [26] deals with the ambiguities of target appearances by learning a low-dimensional subspace representation in increments. MIL [1] handles the ambiguities by employing multiple instances of the appearance. L1 [2, 21] and MTT [35, 34] solve the ambiguities by finding a sparse approximation in a template subspace via L1 minimization. Tracking by detection approaches [8, 10, 13, 27, 30] overcome the ambiguities by using detection power and advanced machine learning algorithms. Different from these approaches, the proposed method numerically measures the modeling uncertainties and explicitly applies them to the visual tracking problem.

## 3. Interval based Bayesian Tracking Approach

In this paper, we formulate the posterior by using the interval representation as follows.

$$[p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) \propto [p]\left(\mathbf{Y}_t|[\mathbf{X}_t]\right) \times \int p\left([\mathbf{X}_t]|[\mathbf{X}_{t-1}]\right)[p]\left([\mathbf{X}_{t-1}]|\mathbf{Y}_{1:t-1}\right) d[\mathbf{X}_{t-1}], \tag{1}$$

where $[p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right)$ denotes the posterior that has an interval. To formulate $[p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right)$ in (1), we should design the state interval, $[\mathbf{X}_t]$, the likelihood interval, $[p](\mathbf{Y}_t|[\mathbf{X}_t])$, and the transition probability, $p([\mathbf{X}_t]|[\mathbf{X}_{t-1}])$. Note that the transition probability is modeled as $p([\mathbf{X}_t]|[\mathbf{X}_{t-1}])$ instead of $[p]([\mathbf{X}_t]|[\mathbf{X}_{t-1}])$, since it is very difficult to design the interval of the state transition [22]. For better understanding, four different types of the posterior are illustrated in Fig.3.

### 3.1. Modeling of $[\mathbf{X}_t]$

The interval representation of the state, $[\mathbf{X}_t]$ is defined by

$$[\mathbf{X}_t] = [\underline{\mathbf{X}}_t, \overline{\mathbf{X}}_t] = [(\underline{X}_t^1, \underline{X}_t^2, \underline{X}_t^3)^T, (\overline{X}_t^1, \overline{X}_t^2, \overline{X}_t^3)^T], \tag{2}$$

where $\underline{\mathbf{X}}_t \leq [\mathbf{X}_t] \leq \overline{\mathbf{X}}_t$ with the element-wise manner and; $X_t^1$, $X_t^2$, and $X_t^3$ indicate x-center, y-center positions, and the scale of the target, respectively.

### 3.2. Modeling of $[p](\mathbf{Y}_t|[\mathbf{X}_t])$

The interval representation of the likelihood, $[p](\mathbf{Y}_t|[\mathbf{X}_t])$ is defined by

$$[p](\mathbf{Y}_t|[\mathbf{X}_t]) = [\underline{[p]}(\mathbf{Y}_t|[\mathbf{X}_t]), \overline{[p]}(\mathbf{Y}_t|[\mathbf{X}_t])], \tag{3}$$

where $\underline{[p]}(\mathbf{Y}_t|[\mathbf{X}_t])$ and $\overline{[p]}(\mathbf{Y}_t|[\mathbf{X}_t])$ are the lower and upper bounds of $[p](\mathbf{Y}_t|[\mathbf{X}_t])$, respectively. Now, $[p](\mathbf{Y}_t|[\mathbf{X}_t])$ in (3) can be decomposed into two terms by the first-order interval Taylor extension [31] w.r.t. a reference state $\dot{\mathbf{X}}_t$. The physical meaning and toy example of the first-order interval Taylor extension is included in the supplementary material.

$$\begin{aligned}
&[p](\mathbf{Y}_t|[\mathbf{X}_t]) \\
&\approx p(\mathbf{Y}_t|\dot{\mathbf{X}}_t) \oplus \sum_{i=1}^{3}\left[\frac{\partial}{\partial X_t^i} p(\mathbf{Y}_t|[\mathbf{X}_t]) \otimes ([X_t^i] \ominus \dot{X}_t^i)\right] \\
&= \underbrace{p(\mathbf{Y}_t|\dot{\mathbf{X}}_t)}_{\text{Single value}} \oplus \underbrace{\left[\sum_{i=1}^{3}\left(\lambda_i(\underline{X}_t^i - \dot{X}_t^i)\right), \sum_{i=1}^{3}\left(\lambda_i(\overline{X}_t^i - \dot{X}_t^i)\right)\right]}_{\text{Interval value}},
\end{aligned} \tag{4}$$

where $\ominus$[1], $\otimes$[2], and $\oplus$ indicate the element-wise minus, time, and plus operations, respectively. In (4), $\lambda_i$ is approximated by

$$\lambda_i \approx \text{MAX}\left(\left|\frac{\partial}{\partial X_t^i} p(\mathbf{Y}_t|[\mathbf{X}_t])\right|\right) > 0, \tag{5}$$

where the approximation is to simplify the interval length in (10). Nevertheless, this approximation is good enough to obtain the accurate tracking results, as demonstrated in the experiments. In (4), $\dot{\mathbf{X}}_t$ can be any point that belongs to $[\mathbf{X}_t]$. In our proposed method, we set $\dot{\mathbf{X}}_t = (\dot{X}_t^1, \dot{X}_t^2, \dot{X}_t^3)$ to the MMSE estimate of $\mathbf{X}_t$ over $[\mathbf{X}_t]$ with respect to $p(\mathbf{Y}_t|[\mathbf{X}_t])$, as follows: $\dot{\mathbf{X}}_t = \arg\min_{\hat{\mathbf{X}}} \mathbb{E}_{p(\mathbf{Y}_t|[\mathbf{X}_t])}\|\mathbf{X}_t - \hat{\mathbf{X}}\|^2$, for $\mathbf{X}_t \in [\mathbf{X}_t]$.

The first term in (4) has a single value, which is defined by

$$p(\mathbf{Y}_t|\dot{\mathbf{X}}_t) = e^{-\gamma_1 Dist(\mathbf{Y}_t(\dot{\mathbf{X}}_t), M_t)}, \tag{6}$$

[1] $[X_t^i] \ominus \dot{X}_t^i = \left[\underline{X}_t^i - \dot{X}_t^i, \overline{X}_t^i - \dot{X}_t^i\right]$.

[2] $\frac{\partial}{\partial X_t^i} p(\mathbf{Y}_t|[\mathbf{X}_t]) \otimes ([X_t^i] \ominus \dot{X}_t^i) = \left[\lambda_i(\underline{X}_t^i - \dot{X}_t^i), \lambda_i(\overline{X}_t^i - \dot{X}_t^i)\right]$.

where $\gamma_1$ denotes the weighting parameter, $\mathbf{Y}_t(\dot{\mathbf{X}}_t)$ represents the observation of the image patch described by $\dot{\mathbf{X}}_t$ and $M_t$ indicates the target model at time $t$. The $Dist$ function returns the distance between the observation $\mathbf{Y}_t(\dot{\mathbf{X}}_t)$ and the target model $M_t$. For example, we can use the HSV color histogram [24] for $\mathbf{Y}_t(\dot{\mathbf{X}}_t)$ and $M_t$, whereas we can employ Bhattacharyya similarity coefficient [24] or diffusion distance [19] for the $Dist$ function. In (4), the second term has an interval value, where $p(\mathbf{Y}_t|[\mathbf{X}_t])$ is defined by

$$\begin{aligned} p(\mathbf{Y}_t|[\mathbf{X}_t]) &= [\underline{p}(\mathbf{Y}_t|[\mathbf{X}_t]), \overline{p}(\mathbf{Y}_t|[\mathbf{X}_t])], \\ \underline{p}(\mathbf{Y}_t|[\mathbf{X}_t]) &= \text{MIN}(\{p(\mathbf{Y}_t|\mathbf{X}_t)|\mathbf{X}_t \in [\mathbf{X}_t]\}), \\ \overline{p}(\mathbf{Y}_t|[\mathbf{X}_t]) &= \text{MAX}(\{p(\mathbf{Y}_t|\mathbf{X}_t)|\mathbf{X}_t \in [\mathbf{X}_t]\}). \end{aligned} \tag{7}$$

### 3.3. Modeling of $p([\mathbf{X}_t]|[\mathbf{X}_{t-1}])$

The transition probability $p([\mathbf{X}_t^*]|[\mathbf{X}_t])$ is realized by the proposal density function $Q([\mathbf{X}_t^*];[\mathbf{X}_t])$. Our proposal density function is different from conventional ones because the function employs the state interval. To handle the state interval, we design three proposal density functions for the $x, y$ positions and the scale, $Q([X_t^{i*}] = [\underline{X}_t^{i*}, \overline{X}_t^{i*}];[X_t^i])$ for $i = 1, 2, 3$. The lower bound $\underline{X}_t^{i*}$ and the upper bound $\overline{X}_t^{i*}$ of the proposed state interval $[X_t^{i*}]$ are thus obtained as follows:

$$\begin{aligned} &\underline{X}_t^{i*} = \text{MIN}(\{X_t^{i*}\}),\ \overline{X}_t^{i*} = \text{MAX}(\{X_t^{i*}\}), \\ &\text{where } \{X_t^{i*}|X_t^{i*} \sim G(X_t^i, \sigma_i), X_t^i \in [X_t^i]\}. \end{aligned} \tag{8}$$

In (8), $G(X_t^i, \sigma_i)$ denotes the Gaussian function with mean $X_t^i$ and variance $\sigma_i$. Then, $\dot{\mathbf{X}}_t^*$ is derived by obtaining the expectation of $[\mathbf{X}_t^*]$ with respect to $p(\mathbf{Y}_t|[\mathbf{X}_t^*])$.

### 3.4. Decomposition of $[p]([\mathbf{X}_t]|\mathbf{Y}_{1:t})$

Now, by inserting (4) into (1), we can decompose the posterior as

$$\begin{aligned} [p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) \approx \underbrace{p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})}_{\text{Single value}} \oplus \\ \underbrace{\alpha \otimes \left[\sum_{i=1}^{3}\left(\lambda_i(\underline{X}_t^i - \dot{X}_t^i)\right), \sum_{i=1}^{3}\left(\lambda_i(\overline{X}_t^i - \dot{X}_t^i)\right)\right]}_{\text{Interval value}}, \end{aligned} \tag{9}$$

where $\alpha = [\underline{\alpha}, \overline{\alpha}] = \int p([\mathbf{X}_t]|[\mathbf{X}_{t-1}])\ [p]([\mathbf{X}_{t-1}]|\mathbf{Y}_{1:t-1})\ d[\mathbf{X}_{t-1}]$. Beacsue $\underline{\alpha}$ and $\overline{\alpha}$ are positive and $\overline{\alpha}$ is larger than $\underline{\alpha}$, the lower and upper bounds of the posterior are then defined by $\underline{[p]}\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) = p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t}) + \underline{\alpha}\ \sum_{i=1}^{3}\left(\lambda_i(\underline{X}_t^i - \dot{X}_t^i)\right)$ and $\overline{[p]}\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) = p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t}) + \overline{\alpha}\ \sum_{i=1}^{3}\left(\lambda_i(\overline{X}_t^i - \dot{X}_t^i)\right)$. The first term in (9) has a single value similar to that of the conventional posterior. The second term in (9) is the interval of the posterior. The interval length of the second term is equal to the gap between the lower and upper bounds of the posterior:

$$\overline{[p]}\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) - \underline{[p]}\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right) \propto \sum_{i=1}^{3}\left(\lambda_i\overline{\alpha}(\overline{X}_t^i - \underline{X}_t^i)\right). \tag{10}$$

Note that the large interval length means that the posterior $[p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right)$ has a large modeling error. Therefore, as a measure for the modeling accuracy of the posterior, we define the following probability:

$$F([p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right)) = e^{-\gamma_2 \sum_{i=1}^{3}\left(\lambda_i\overline{\alpha}(\overline{X}_t^i - \underline{X}_t^i)\right)}, \tag{11}$$

where $\sum_{i=1}^{3}\left(\lambda_i\overline{\alpha}(\overline{X}_t^i - \underline{X}_t^i)\right)$ in (10) is the interval length of the posterior and $\gamma_2$ is the weighting parameter. If the interval length is large, the probability in (11) is close to 0; otherwise, close to 1. The two terms in (9) induce two different visual trackers in the next section.

## 4. The M4 Estimation

Now, let us define our goal of visual tracking as to find the best state interval $[\hat{\mathbf{X}}_t]$ that maximizes the posterior and simultaneously minimizes modeling uncertainty of the posterior. To achieve this goal, our method should search $[\hat{\mathbf{X}}_t]$ that maximizes $p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})$ in (9) and minimizes $\alpha \otimes \left[\sum_{i=1}^{3}\left(\lambda_i(\underline{X}_t^i - \dot{X}_t^i)\right), \sum_{i=1}^{3}\left(\lambda_i(\overline{X}_t^i - \dot{X}_t^i)\right)\right]$ in (9) by maximizing $F([p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right))$ in (11). This can be formulated by

$$[\hat{\mathbf{X}}_t] \equiv \underset{[\mathbf{X}_t]}{\arg\max}\, \omega_1 p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t}) + \omega_2 F([p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right)), \tag{12}$$

where $p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})$ in (9) indicates a conventional posterior; $F([p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right))$ in (11) denotes the modeling accuracy of the posterior; and $\omega_1$ and $\omega_2$ are the weighting parameters, which are automatically determined in Section 4.3. We can interpret (12) in terms of two trackers, such that $p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})$ induces one tracker, while $F([p]\left([\mathbf{X}_t]|\mathbf{Y}_{1:t}\right))$ represents another tracker. To obtain a solution of (12), the first tracker searches for the state interval that maximizes the posterior by using the MMSE-MAP estimation, while the second tracker searches for the state interval that minimizes the interval length of the posterior by using the ML-MIL estimation. Then the Interacting Markov Chain Monte Carlo (IMCMC) sampling method in [6, 14, 15] finds the best common interval of the state for the two trackers via the interaction between them. Note that the summation of two terms in (12) is derived from (9), wherein the posterior is decomposed into the summation of two terms.

### 4.1. Tracker 1: MMSE-MAP Estimation

The goal of tracker 1 is to find the best state interval that maximizes the first term in (12), $p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})$. Since our method maximizes the posterior over the MMSE state $p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})$, this estimation is called MMSE-MAP. To achieve the best MMSE-MAP state, our method obtains samples over Markov Chain 1 via two steps: the proposal and acceptance steps. In the proposal step, a new state interval, $[\mathbf{X}_t^*]$, is proposed based on the previous state interval, $[\mathbf{X}_t]$, by using the proposal density function $Q([\mathbf{X}_t^*];[\mathbf{X}_t])$ in (8). Then, $\dot{\mathbf{X}}_t^*$ is obtained by $\mathbb{E}_{p(\mathbf{Y}_t|[\mathbf{X}_t^*])}[\mathbf{X}_t^*]$. Given the proposed state interval, the chain decides whether the proposed state interval is accepted or not with the acceptance ratio in the acceptance step. The acceptance ratio is designed toward accepting the state interval, which maximizes the posterior over the MMSE state:

$$a_1^p = min\left[1, \frac{p(\dot{\mathbf{X}}_t^*|\mathbf{Y}_{1:t})Q([\mathbf{X}_t];[\mathbf{X}_t^*])}{p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})Q([\mathbf{X}_t^*];[\mathbf{X}_t])}\right]. \quad (13)$$

These two steps proceed iteratively until the number of iterations reaches a predefined value.

### 4.2. Tracker 2: ML-MIL Estimation

The goal of tracker 2 is to find the best state interval that maximizes the second term in (12), $F([p]([\mathbf{X}_t]|\mathbf{Y}_{1:t}))$. To maximize $F([p]([\mathbf{X}_t]|\mathbf{Y}_{1:t}))$ in (11), both the interval length of the state, $\overline{\alpha}(\overline{X}_t^i - \underline{X}_t^i)$, and $\lambda_i \approx \text{MAX}\left(\left|\frac{\partial}{\partial X_t^i}p(\mathbf{Y}_t|[\mathbf{X}_t])\right|\right)$ in (5) should be minimized.

- The state inteval that has a minimum interval length is chosen, which is called the MIL estimation.
- To minimize $\lambda_i$, a derivative of the likelihood should be minimized by the ML estimation.

Similarly to the Tracker 1, the best ML-MIL state can be achieved by obtaining samples over Markov Chain 2 via two steps: the proposal and acceptance steps. The proposal step is the same as that in tracker 1, where the proposal density function $Q([\mathbf{X}_t^*];[\mathbf{X}_t])$ is used. The acceptance step is designed to accept the samples of state interval frequently that minimize both the interval length of the state and the derivative of the likelihood:

$$a_2^p = min\left[1, \frac{F(p([\mathbf{X}_t^*]|\mathbf{Y}_{1:t}))Q([\mathbf{X}_t];[\mathbf{X}_t^*])}{F(p([\mathbf{X}_t]|\mathbf{Y}_{1:t}))Q([\mathbf{X}_t^*];[\mathbf{X}_t])}\right]. \quad (14)$$

### 4.3. Interaction between Two Trackers

To obtain the best state interval that satisfies (12), the two trackers designed in the above subsections are integrated in an IMCMC framework, that consists of two modes, parallel and interacting. By using the two modes separately, the landscapes of the decomposed posteriors can be simplified and, thus, sampling methods prevent the Markov Chains from getting trapped in local minima and efficiently search for the best state [20]. In the parallel mode, the method acts as the parallel Metropolis Hastings algorithms by using (8) and (13) for the Tracker 1, and (8) and (14) for the Tracker 2. When the method is in the interacting mode, the two trackers communicate with each other and leap to better state intervals of the target. In the interacting mode, the two trackers accept the state interval of tracker 1, $[\mathbf{X}_t]_1$, as their next state intervals with the following probability:

$$a_1^i = \frac{p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t})}{p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t}) + F(p([\mathbf{X}_t]_2|\mathbf{Y}_{1:t}))}, \quad (15)$$

where $\dot{\mathbf{X}}_t = \mathbb{E}_{p(\mathbf{Y}_t|[\mathbf{X}_t]_1)}[\mathbf{X}_t]_1$. Similarly, the two trackers accept the state interval of tracker 2, $[\mathbf{X}_t]_2$, as their next state intervals with the following probability:

$$a_2^i = \frac{F(p([\mathbf{X}_t]_2|\mathbf{Y}_{1:t}))}{p(\dot{\mathbf{X}}_t|\mathbf{Y}_{1:t}) + F(p([\mathbf{X}_t]_2|\mathbf{Y}_{1:t}))}. \quad (16)$$

Based on the interaction between the two trackers, common states can be evaluated by both the Tracker 1 and 2. Our method operates in an interacting mode with the probability $\beta$, which linearly decreases from 1.0 to 0.0 as the simulation continues. During the interaction process, $\omega_1$ and $\omega_2$ in (12) are implicitly determined as follows:

$$\omega_1 = \frac{\text{Number of accepted states by tracker 1}}{\text{Number of proposed states by two trackers}}, \quad \omega_2 = \frac{\text{Number of accepted states by tracker 2}}{\text{Number of proposed states by two trackers}}. \quad (17)$$

The supplementary material includes the entire procedure of the proposed method.

## 5. Experimental Results

The proposed method (Interval Tracker:IT) was compared with 8 recent tracking methods: IVT [26], MILT [1], L1T [2, 21], MTT [35], VTD [14], VTS [15], TLD [13], and MUG [16]. The parameters of each tracker were adjusted to produce the best tracking results, whereas our method utilized the *fixed parameter setting* in all the experiments. $\gamma_1$ in (6) and $\gamma_2$ in (11) were set to 5 for all the experiments. $\sigma_1$, $\sigma_2$, and $\sigma_3$ in (8) were set to $\sigma_1 = 2, \sigma_2 = 1.414$, and $\sigma_3 = 0.0165$ for all experiments. For the sampling-based tracking methods, we used the same number of samples, 800. For the other methods, we used the authors' codes. The tracking methods were evaluated using a total of 15 challenging sequences [3]. The result video, data, and code are available at http://cv.snu.ac.kr/research/IT.

[3] $singer1^L$ and $skating1^L$ are obtained from [17], which are the low frame rate version of *singer1* and *skating1* in [14], respectively.

### 5.1. Implementation Details

**Initialization:** At the initial frame, the bounding box is drawn manually over the target region, which determines the $x, y$ positions $(X^1, X^2)$ and the scale $X^3$ of the target. Then, the initial state interval $[\mathbf{X}_0] = \left[(\underline{X}_0^1, \underline{X}_0^2, \underline{X}_0^3)^T, (\overline{X}_0^1, \overline{X}_0^2, \overline{X}_0^3)^T\right]$ is set to $\underline{X}_0^1 = X^1 - 0.25B_w, \overline{X}_0^1 = X^1 + 0.25B_w, \underline{X}_0^2 = X^2 - 0.25B_h, \overline{X}_0^2 = X^2 + 0.25B_h, \underline{X}_0^3 = X^3 - 0.05$, and $\overline{X}_0^3 = X^3 + 0.05$, where $B_w$ and $B_h$ denote the width and the height of the initial bounding box respectively. The target model $M_0$ is made by the HSV histogram using the image patch inside the bounding box. At the beginning of each frame, $[\mathbf{X}_t]$ and $M_t$ are initialized in the same manner based on the best state at the previous frame, $\hat{\mathbf{X}}_{t-1}$.

**Final Representation:** At each frame, the best state of the target, $\hat{\mathbf{X}}_t$, is represented as

$$\hat{\mathbf{X}}_t = \mathbb{E}_{p(\mathbf{Y}_t|[\mathbf{X}_t])}[\hat{\mathbf{X}}_t], \tag{18}$$

where $[\hat{\mathbf{X}}_t]$ indicates the best state interval, which is found by (12). The final representation of the target state in (18) enables our tracking results to be evaluated and to be compared with other tracking methods. This final representation can be justified empirically. The interval length of the state decreases and usually converges into a single state, as demonstrated in the convergence part of Section 5.2. The target model $M_t$ in (6) can then be updated at each time $t$ by combining the 5 recent image patches (e.g. image patch at time $t$ is described by the best state $\hat{\mathbf{X}}_t$ ) with the initial image patch.

**Approximation:** To estimate $\dot{X}_t^i$ in (4) and $\lambda_i$ in (5) and to get $\underline{X}_t^{i*}$ and $\overline{X}_t^{i*}$ in (8), our method should consider all $X_t^i \in [X_t^i]$. Since it is intractable to consider all $X_t^i$ in $[X_t^i]$, our method samples the 10 numbers of $X_t^i$ in $[X_t^i]$ and approximately obtains $\dot{X}_t^i$, $\lambda_i$, $\underline{X}_t^{i*}$, and $\overline{X}_t^{i*}$. Our method also obtains an approximate derivative of the likelihood function, $\frac{d}{d\mathbf{X}_t} p(\mathbf{Y}_t|[\mathbf{X}_t])$, in (5) by using finite differences.

### 5.2. Analysis of the Proposed Method

**Plug-in:** Our method is highly applicable because it can be easily combined with other original tracking methods and can greatly improve their tracking performance. The Center Location Error (CLE) of VTD [14] decreased from 72 to 9 when VTD is combined with our method. The CLE of VTS [15] decreased from 71 to 15 when VTS is combined with our method. Original VTD is worse than VTS because VTD uses a fixed number of trackers. However, our method changes VTD to use a varying number of trackers. The large interval length of the posterior means that VTD uses a large number of trackers. Hence, by combining our method, VTD can be more enhanced than VTS. The speed of our method depends on the speed of the original methods. For example, our method combined with takes $1 \sim 5$ seconds per frame. The plug-in process replaces each Markov Chain of the original methods with two Markov Chains, as discussed in Sections 4.1 and 4.2. As an example, the plug-in process with VTD is described in the supplementary material.

Table 1. **Tracking results with several estimation methods**. The numbers indicate average CLEs in pixels. The improvement is the error difference between two neighbor steps. IT-VTD denotes our method combined with VTD.

| | A step:MAP | B:A+MMSE | C:B+MIL | D:C+ML |
|---|---|---|---|---|
| IT-VTD | 72 | 59 | 31 | 9 |
| Improvement | N/A | 13 | **28** | 22 |

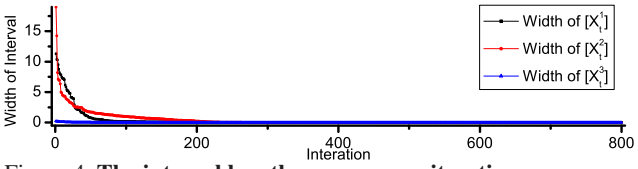



Figure 4. **The interval length converges as iteration goes on.**

**Fusion of Several Estimation Methods:** Our method can have the aforementioned advantages and can accurately track the targets because it efficiently fused several estimation methods. Table 1 describes the role each estimation method plays to improve the tracking accuracy. For example, in the C step, our method fused MAP, MMSE, and MIL. By additionally inserting MIL into the B step, the 28 tracking errors was reduced. Our method greatly enhanced the tracking accuracy by employing MIL, which demonstrates that MIL makes the overall algorithm successful. Introducing MIL into the estimation process is significant because the modeling of the posterior cannot be perfect, and thus, the modeling error should be considered.

**Convergence:** Our method has a real solution by decreasing the interval length of the state during the tracking process, although the method starts from the interval. This is also why the best solution can be represented by a single state in (18) instead of interval. Fig.4 empirically demonstrates that the interval length of the state decreases and usually converges into a single state as time passes. The IMCMC [6] algorithm also makes our method converge, although the method fuses four estimation methods and combines two posteriors constructed by two trackers.

### 5.3. Comparison with State-of-the-Art Methods

Tables 2 and 3 demonstrate that our method combined with VTD, IT-VTD is the best in terms of tracking accuracy. For this experiment, several state-of-the-art tracking methods were compared using challenging benchmark dataset and our dataset. Other tracking methods showed good tracking performance, but when the target appearance and state were highly ambiguous in either the initial step or during the tracking process, these methods failed to accurately track the targets. Note that the success rate results

Table 2. **Comparison of tracking results using the benchmark dataset.** The numbers indicate average CLEs in pixels and success rates. Red is the best result and blue is the second-best result. Because TLD didn't produce tracking results for some frames, we calculated average CLEs when TLD produced the results for more than 10 percentage of whole frames.

| | IVT | MILT | L1T | MTT | VTD | VTS | MUG | TLD | IT-VTD |
|---|---|---|---|---|---|---|---|---|---|
| *car4* | **3**(**95**) | 50(34) | **3**(85) | **3**(**95**) | 35(39) | 123(38) | 20(60) | 10(70) | 6(77) |
| *coke* | 30(31) | 21(37) | 29(31) | **5**(**90**) | 43(17) | 34(17) | 15(42) | 23(35) | **18**(**39**) |
| *david* | **4**(**98**) | 23(29) | 19(31) | 7(70) | 7(35) | 7(55) | 21(33) | **5**(**98**) | **5**(97) |
| *girl* | 24(43) | 32(41) | 23(43) | **5**(**75**) | 16(50) | 16(50) | 28(42) | 11(50) | **9**(**51** ) |
| *shaking* | 95(22) | 38(82) | 66(22) | 9(87) | **5**(90) | **5**(**98**) | 25(20) | **5**(**98**) | **4**(**99**) |
| $singer1^L$ | **8**(**91**) | 29(32) | **5**(**98**) | 45(33) | 11(76) | 25(33) | 9(89) | 13(21) | 10(69) |
| $skating1^L$ | 160(22) | 64(37) | 78(26) | 63(33) | **8**(**87**) | **9**(82) | 12(75) | 195(19) | **8**(**84**) |
| *soccer* | 151(20) | 41(26) | 40(26) | **17**(**34**) | 21(32) | **15**(**35**) | 32(20) | N/A | 21(33) |
| *sylv* | 48(75) | 11(87) | 5(96) | 5(96) | 21(70) | 15(80) | **4**(**98**) | 6(97) | **3**(**99**) |
| *tiger1* | 65(30) | 15(65) | 23(37) | 28(34) | 13(69) | **6**(80) | 25(50) | **6**(**81**) | **4**(**97**) |
| *tiger2* | 47(23) | 17(70) | 26(30) | 23(40) | 45(23) | 26(33) | **12**(**76**) | 14(74) | **6**(**80**) |
| average | 57(50) | 31(49) | 28(47) | 19(62) | 20(53) | 25(54) | **18**(55) | 28(**64**) | **8**(**75**) |

Table 3. **Comparison of tracking results using our new dataset.**

| | IVT | MILT | L1T | MTT | VTD | VTS | MUG | TLD | IT-VTD |
|---|---|---|---|---|---|---|---|---|---|
| *mission* | 201(20) | 171(19) | 192(19) | 229(22) | 201(18) | **164**(**19**) | 175(18) | N/A | **11**(**98**) |
| *penguin* | 54(45) | 249(16) | 68(33) | **16**(**65**) | 129(12) | 95(12) | 17(63) | 186(15) | **11**(**88**) |
| *rhinoceros* | 214(14) | 238(12) | 156(14) | 210(16) | 208(15) | 224(15) | 238(15) | 170(16) | **3**(**98**) |
| *terminator* | 236(12) | 328(**17**) | 140(12) | **104**(13) | 318(13) | 308(13) | 230(14) | N/A | **10**(**93**) |
| average | 176(22) | 246(16) | **139**(19) | **139**(**29**) | 214(14) | 197(14) | 165(27) | 178(15) | **8**(**94**) |

were consistent with the center location results. A high CLE but low success rate produced by a few tracking methods means that they are weak to handle scale changes.

Fig.5 shows the qualitative tracking results of several state-of-the-art tracking methods. In Fig.5(a) to 5(d), the initial target models at frame 1 were severely corrupted by occlusions and illumination changes. Nevertheless, our method (yellow boxes) robustly tracked the targets in the following frames. Other methods such as MTT, VTS, and MILT failed to track the targets in the following frames due to this ambiguous initialization. In Fig.5(e) to 5(h), our method tracked the targets more accurately than other methods, even though the sequences include real-world tracking scenarios such as illumination changes, abrupt motions, and occlusions. In Fig.5(i) to 5(m), our method successfully tracked the targets on the widely used benchmark datasets.

## 6. Conclusion and Discussion

To solve the visual tracking problem, we propose the M4 estimation, which combines MAP, MMSE, ML, and MIL estimations. In the M4 estimation, we represent the posterior as an interval and explicitly measure the modeling error of the posterior. Then, we find the best state, which maximizes a posterior and, at the same time, minimizes interval of the posterior.

Our method uses the curvature information of the posterior to measure uncertainty. The curvature of the posterior is certainly related to the uncertainty. The flat curvature of the posterior within a state interval means that posterior values within the interval are confidently estimated because neighbor states agree about the values. Because our method searches for a high posterior value as well as the flat posterior, the method can find the MAP solution with small uncertainty.

Our method obtains the uncertainty by using posterior itself. Hence other trackers can be easily integrated into our method without any adaptation of outside sources. Because the outside sources are not available in some cases, our approach is more applicable. In our method, this integration can be performed by transforming the standard sequential Bayesian filtering to its interval version. Hence our integration is far from naive post-processing.

Considering the flat curvature of the posterior can reduce discriminativeness and make it harder to find the optimal point. To alleviate this problem, our method searches a good state in the flat region of the posterior by using multiple criteria (i.e. MAP, MMSE, ML, and MIL).

There could be other simpler methods (e.g. [5]) to achieve the same goal with ours. However, the reason why we followed the strategy in [22] is to get the *optimal* uncertainty in terms of the interval analysis. As addressed in [22], our interval forms the mathematical lower and upper bounds of the posterior. Actually, we have tested simpler approaches without the interval analysis, but we could not get better results.

## References


[1] B. Babenko, M. Yang, and S. Belongie. Visual tracking with online multiple instance learning. *CVPR*, 2009. 3, 5

[2] C. Bao, Y. Wu, H. Ling, and H. Ji. Real time robust l1 tracker using accelerated proximal gradient approach. *CVPR*, 2012. 3, 5

[3] A. Benavoli, M. Zaffalon, and E. Miranda. Robust filtering through coherent lower previsions. *IEEE Trans. Automat. Contr.*, 56(7):1567–1581, 2011. 1, 2

[4] R. T. Collins, Y. Liu, and M. Leordeanu. Online selection of discriminative tracking features. *PAMI*, 27(10):1631–1643, 2005. 1

[5] D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *PAMI*, 25(5):564–575, 2003. 7

[6] J. Corander, M. Ekdahl, and T. Koski. Parallell interacting MCMC for learning of topologies of graphical models. *Data Min. Knowl. Discov.*, 17(3), 2008. 4, 6

[7] M. Godec, P. M. Roth, , and H. Bischof. Hough-based tracking of non-rigid objects. *ICCV*, 2011. 1

[8] C. L. H. Grabner and H. Bischof. Semi-supervised on-line boosting for robust tracking. *ECCV*, 2008. 3

[9] B. Han and L. Davis. On-line density-based appearance modeling for object tracking. *ICCV*, 2005. 1

[10] S. Hare, A. Saffari, and P. H. S. Torr. Struck: Structured output tracking with kernels. *ICCV*, 2011. 3

[11] S. He, Q.-X. Yang, R. Lau, J. Wang, and M.-H. Yang. Visual tracking via locality sensitive histograms. *CVPR*, 2013. 1

[12] A. D. Jepson, D. J. Fleet, and T. F. E. Maraghi. Robust online appearance models for visual tracking. *PAMI*, 25(10):1296–1311, 2003. 1

[13] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *PAMI*, 34(7):1409–1422, 2012. 3, 5

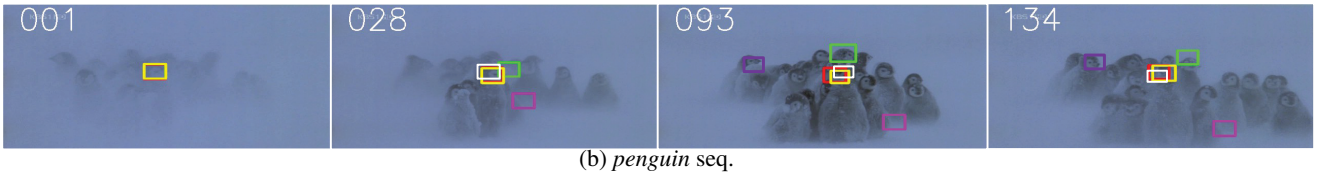
[14] J. Kwon and K. M. Lee. Visual tracking decomposition. *CVPR*, 2010. 2, 4, 5, 6

[15] J. Kwon and K. M. Lee. Tracking by sampling trackers. *ICCV*, 2011. 2, 4, 5, 6

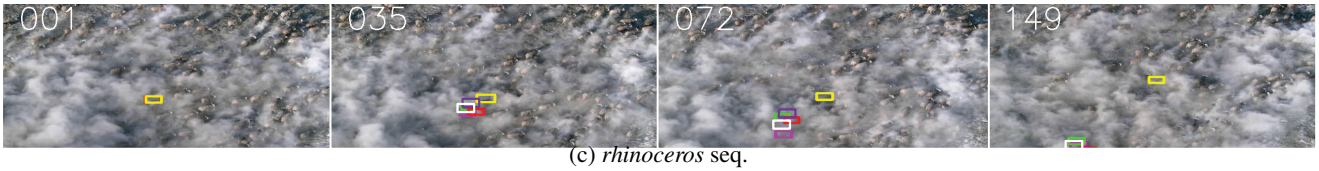
[16] J. Kwon and K. M. Lee. Minimum uncertainty gap for robust visual tracking. *CVPR*, 2013. 2, 3, 5

[17] J. Kwon and K. M. Lee. Tracking by sampling and integrating multiple trackers. *TPAMI*, 2013. 5


(a) *mission* seq.

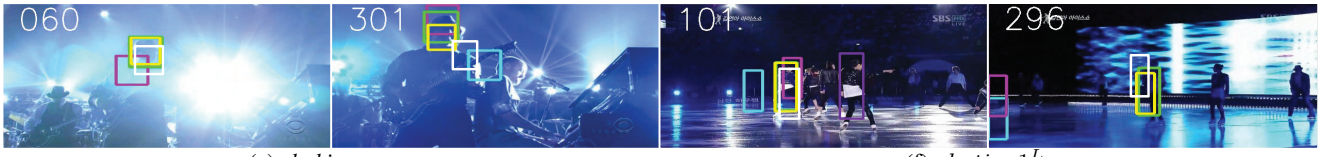



(b) *penguin* seq.

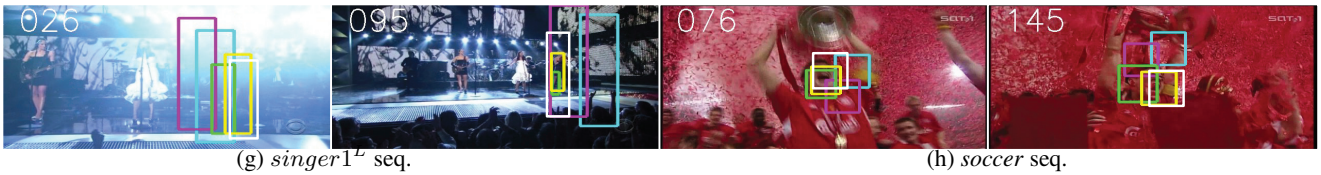



(c) *rhinoceros* seq.


(d) *terminator* seq.




(e) *shaking* seq. (f) $skating1^L$ seq.




(g) $singer1^L$ seq. (h) *soccer* seq.


(i) *coke* seq. (j) *david* seq. (k) *girl* seq. (l) *tiger2* seq. (m) *car* seq.

Figure 5. **Qualitative comparison of the tracking results using other methods.** The yellow, red, green, pink, blue, white, and violet boxes represent the tracking results of IT-VTD, MTT, VTS, MILT, FRAGT, MUG, and TLD, respectively.


[18] X. Li, A. Dick, H. Wang, C. Shen, and A. van den Hengel. Graph mode-based contextual kernels for robust svm tracking. *ICCV*, 2011. 1

[19] H. Ling and K. Okada. Diffusion distance for histogram comparison. *CVPR*, 2006. 4

[20] D. J. C. Mackay. Introduction to monte carlo methods. *In Learning in Graphical Models, M. I. Jordan, Ed. NATO Science Series. Kluwer Academic Press*, 1998. 5

[21] X. Mei and H. Ling. Robust visual tracking and vehicle classification via sparse representation. *PAMI*, 33(11):2259–2272, 2011. 3, 5

[22] R. E. Moore. Interval analysis. *Prentice-Hall*, 1966. 2, 3, 7

[23] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan. Locally orderless tracking. *CVPR*, 2012. 1

[24] P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. *ECCV*, 2002. 4

[25] A. E. Raftery and Y. Zheng. Discussion: Performance of bayesian model averaging. *J. Amer. Statistical Assoc.*, 98, 2003. 2

[26] D. A. Ross, J. Lim, R. Lin, and M. Yang. Incremental learning for robust visual tracking. *IJCV*, 77(1):125–141, 2008. 3, 5

[27] J. Santner, C. Leistner, A. Saffari, T. Pock, and H. Bischof. Prost: Parallel robust online simple tracking. *CVPR*, 2010. 3

[28] L. Sevilla-Lara and E. Learned-Miller. Distribution fields for tracking. *CVPR*, 2012. 1

[29] K. Smith, D. Gatica-Perez, and J.-M. Odobez. Using particles to track varying numbers of interacting people. *CVPR*, 2005. 1

[30] S. Stalder, H. Grabner, and L. V. Gool. Cascaded confidence filtering for improved tracking-by-detection. *ECCV*, 2010. 3

[31] G. Trombettoni, I. Araya, B. Neveu, and G. Chabert. Inner regions and interval linearizations for global optimization. *AAAI*, 2011. 2, 3

[32] M. Yang and Y. Wu. Tracking non-stationary appearances and dynamic feature selection. *CVPR*, 2005. 1

[33] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4), 2006. 1

[34] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Low-rank sparse learning for robust visual tracking. *ECCV*, 2012. 3

[35] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. *CVPR*, 2012. 3, 5