Target Tracking with Online Feature Selection in FLIR Imagery

Vijay Venkataraman, Guoliang Fan and Xin Fan
School of Electrical and Computer Engineering
Oklahoma State University, Stillwater, OK 74078
{vijay.venkataraman,guoliang.fan,xin.fan}@okstate.edu

Abstract

We present a particle filter-based target tracking algorithm for FLIR imagery. A dual foreground and background model is proposed for target representation which supports robust and accurate target tracking and size estimation. A novel online feature selection technique is introduced that is able to adaptively select the optimal feature to maximize the tracking confidence. Moreover, a coupled particle filtering approach is developed for joint target tracking and feature selection in an unified Bayesian estimation framework. The experimental results show that the proposed algorithm can accurately track poorly-visible targets in FLIR imagery even with strong ego-motion. The tracking performance is improved when compared to the tracker with a foreground-based target model and without online feature selection.

1. Introduction

Target tracking in the forward looking infrared (FLIR) imagery has remained a challenging problem. Usually, FLIR images are characterized by low signal-to-noise (SNR) ratios, poor target visibility, and time varying target appearances. These factors call for special treatments in FLIR tracking algorithms. In addition strong ego motion of the FLIR sensor makes it difficult to characterize the target’s kinematics. Two interrelated issues will be discussed in this paper: (1) how to develop an elaborate target appearance model for superior tracking performance and (2) how to select optimal features to take advantage of the new target model. Despite the fact that size estimation is usually important, especially in automatic missile guidance systems where the size may act as a cue for distance, it is usually not considered in most FLIR tracking algorithms [12], partially because of the lack of robust and reliable target appearance model. For example, to achieve a precise position localization, traditional tracking algorithms assign less confidence to the target’s boundary pixels compared with the center ones [9, 12]. However, this biased confidence assignment reduces the sensitivity for size estimation.

In this paper, we propose a dual appearance model that accounts for both foreground and background appearances. Unlike the classifier-based tracking [4, 8] approach, we track the target by matching both its foreground and background statistics with their corresponding reference models. These statistics are used as features for appearance modeling. As the target appearance changes with time, the contribution of each feature towards tracking will vary. It is desired to provide a feature selection scheme that can adaptively assign different weights to different features indicating their relative importance to the tracking process. We formulate the online weight update as a probabilistic estimation problem that is directly integrated with the tracker. In addition to accurate target localization and size estimation, the proposed algorithm also selects the optimal feature combination to maximize the confidence of target tracking.

Tracking algorithms in general can be categorized into two major groups namely, target representation and localization algorithms and filtering and data association algorithms [6]. The former is a bottom up process which tries to localize the target based on its appearance, such as mean-shift tracking. The latter is a top down approach which relies on prediction and update of the target’s probability density functions, such as Kalman filtering. Methods also exist which combine the two approaches into one framework [9]. In our work, we follow the filtering and data association framework where a coupled particle filtering approach is proposed to jointly track target’s states and to select optimal features for appearance modeling.

The rest of this paper is organized as follows. In Section 2, we review related works and compare the proposed method with existing ones. In Section 3, we present the systematic problem formulation in the framework of sequential Monte Carlo or particle filters. In Section 4, we introduce the kinematic models for target’s position and size. In Section 5, we discuss the proposed dual foreground and background model. In Section 6, a new online feature selection is proposed, and the complete tracking algorithm is also given. Experimental results are shown in Section 7, and conclusions are drawn in Section 8.
2. Related Work

One of the most important issues in target tracking is to model the target’s appearance. In the context of FLIR tracking, the features commonly used for appearance modeling include edges, shapes [10] and local statistics, e.g., intensity or standard deviation (stdev) values [12]. Due to their invariance to scale and slow varying nature, intensity and stdev histograms are widely used for target representation [6, 9, 12]. In our work, we propose a dual target appearance model that includes the intensity and stdev histograms of both the foreground and the background. The use of background information for target tracking was discussed in many tracking algorithms, [2, 3, 4, 8]. In these methods, target tracking is performed on an intermediate image called a confidence map [2], a likelihood image [3] or a weighted image [4] where the pixels are assigned with the probabilities belonging to the background or the foreground. Here we have a different point of view using background for target modeling. It was found that the background would be of great assistance in localizing the target and determining its size. Therefore, we include the background statistics in the target’s appearance model along with that of the foreground.

Feature selection is another important issue for target tracking, e.g., [4, 3]. Usually, feature selection is posed as a class discrimination problem, where the feature that best discriminate foreground from background is selected as the optimal one. Differently, we choose features that maximize the confidence of state estimation for target tracking. In other words, the feature selection criterion is consistent with that of target tracking. Therefore, both feature selection and target tracking can be unified into one probabilistic framework.

Another problem frequently encountered in FLIR tracking is the strong ego motion of the sensor. In [7], two separate global motion compensation modules were involved with two separate tracking modules. In [12], the optical flow computed from the Gabor filter responses is used to compensate the global motion. In this work, we incorporate two dynamic models in the target’s kinematics that are able to deal with ego motion, the target’s motion, and the target’s size variability, and thereby achieve effective and efficient state estimation using the particle filter.

3. Probabilistic formulation

In this section, we study the problem formulation. Let $x_k$ denote the state vector to be estimated that includes the position and size at time instant $k$. In addition, $v_k$ represents the optimal feature set that is associated with the appearance model and will be estimated online. Therefore, target tracking and feature selection are formulated as a state space problem where we need to estimate posterior densities $p(x_k|y_{1:k})$ and $p(v_k|y_{1:k})$ given the observations $y_{1:k}$.

The conditional dependencies between the variables are graphically depicted in Fig. 1. Noticing these dependencies, the estimation can be obtained by recursive Bayesian filters. For $p(x_k|y_{1:k})$, we have

$$p(x_k|y_{1:k}) \propto \int_{v_{k-1}} p(y_k|x_k, v_{k-1})p(v_{k-1}|y_{1:k-1})dv_{k-1} \cdot \int_{x_{k-1}} p(x_k|x_{k-1})p(x_{k-1}|y_{1:k-1})dx_{k-1}. \quad (1)$$

We define the kinematic model of the target, $p(x_k|x_{k-1})$ in Section 4 and the weighted likelihood $p(y_k|x_k, v_{k-1})$ in Section 5. Given $p(x_k|y_{1:k})$, the posterior density of the feature weights $p(v_k|y_{1:k})$ can be represented by:

$$p(v_k|y_{1:k}) = \int_{x_k} p(v_k|x_k, y_{1:k})p(x_k|y_{1:k})dx_k. \quad (2)$$

Let

$$L(v_k) = \int_{x_k} p(x_k|y_{1:k})p(v_k|x_k)p(y_k|v_k, x_k)dx_k, \quad (3)$$

where the weighted likelihood $p(y_k|v_k, x_k)$ has the same form as $p(y_k|v_{k-1}, x_k)$ and $p(v_k|x_k)$ describes how likely the feature weights fit the given tracking estimation, which will be defined in Section 6. Thus (2) becomes

$$p(v_k|y_{1:k}) = L(v_k) \int_{v_{k-1}} p(v_k|v_{k-1})p(v_{k-1}|y_{1:k-1})dv_{k-1}. \quad (4)$$

where $p(v_k|v_{k-1})$ is the evolution prior of the feature weights to be discussed in Section 6.

Motivated by the idea of particle filters [1], we approximate the posterior densities $p(x_k|y_{1:k})$ and $p(v_k|y_{1:k})$ using two weighted sample sets $\{x^i_k, v^i_k\}_{i=1}^{N_p}$ and $\{v^i_k, w^i_k\}_{i=1}^{N_v}$. The integrals in (1) and (4) can be approximated by summations. To avoid the high computational expense brought by integrating $v_{k-1}$ and $x_k$ out from (1) and (3), we replace the variables $v_{k-1}$ and $x_k$ by their expectations $E(v_{k-1})$ and $E(x_k)$ estimated from the weighted sample sets. The detailed algorithm is summarized in Table 1.
4. Kinematic models

In our formulation the state vector at any time instant $k$ is defined as $x_k=[x_k, y_k]$, where $x_k=[x_k, y_k]$ contains the position information and $s_k=[s_k^x, s_k^y]$ shows the size in $x$ and $y$ directions. The foreground area $N_F(x_k)$ is defined a rectangle box whose top-left corner corresponds to coordinate $(x_k, y_k)$ and the length and width are given by $s_k^x$ and $s_k^y$ respectively. In the following, we analyze the position and size dynamics of the target based on the ground truth data, and design appropriate dynamic models for both position and size variations. These dynamic models determine the state transition probabilities, i.e., $p(x_k|x_{k-1})$ that play an important role in particle filtering.

4.1. Target position

First let us consider the dynamics of the target’s position. Ref. 2 shows the evolution of the position and size of the sequences LW-14-15 and LW-15-NS from the AMCOM dataset\(^1\). From Fig. 2 (a) and (b) we observe that for the LW-15-NS sequence, the target stays around the center of the image. This is due to the fact that the sensor is mounted on an airborne platform which is homing in on the target. However the sequence LW-14-15 is characterized by strong ego-motion of the sensor platform. Therefore, we need a model which can account for the low variability of the position when there is no ego motion and at the same time provide more variability when there is strong ego motion. An adaptive framework which can adjust the variability of the position is apt to deal with this requirement.

We employ a first order model which can adapt to the change in the variability of the target state based on the idea in [9]. Such a model requires an estimate of the velocity of the target based on the previous $n$ frames. The velocity at any time instant $k$ is given by equation (5). In the cases when $k < n$ the estimate is made based on all available frames up to time $k$.

$$E[n] = \frac{1}{n} \sum_{l=k-n}^{k-1} |x_l - x_{l-1}|.$$  

Then the state transition model for the position vector $x_k$ is defined as

$$x_k = x_{k-1} + C_k v_k,$$  

where $C_k \propto E[n] \triangle x_k$ and $v_k \sim N(0, I)$. In this model if the target is moving with a low velocity then the variance of the process noise is low, thereby reducing the variability of the target state and vice versa. This increased variability physically reflects the spread of the particles over a larger area in the state space, thereby increasing the probability of locating the target whose position is affected by strong ego motion of the sensor.

4.2. Target size

As the next step we now consider the dynamics of the target size. From Fig. 2 (c) and (d) we observe that the target size has a tendency to increase in steps over time. Though the LW-15-NS sequence shows gradual increase in size, the sequence LW-14-15 is characterized by rapid size changes. In the design of a model for the size dynamics we do not favor an adaptive variance model due to size increments in single frames. Therefore for the size dynamics we employ a simple first order model with fixed variance. The state transition model for the size vector $s_k$ is defined in (7),

$$s_k = s_{k-1} + D w_k,$$  

where $D$ is a fixed constant, and $w_k \sim N(0, I)$. The state transition probabilities can be derived from (6) and (7).

5. Target appearance model

We now discuss the issue of target representation. This model describes the appearance of the target in the image in relation to the underlying states $x_k = [x_k, y_k, s_k^x, s_k^y]$ and therefore defines the likelihood $p(y_k|x_k, v_k)$. We characterize our target using a non-parametric model based on the intensity and stdev histograms of the target area and its local background. We call it the “dual foreground-background model” since in addition to the information from the target area we also model its background.

---

\(^1\)http://cis.jhu.edu/data.sets/AMCOM/amcom.html
5.1. Dual foreground-background model

At any given time instant \(k\) assume the intensity image \(I_k\) and the stdev image \(S_k\) are available. Given \(x_k\), we can determine foreground area \(N_F(x_k)\) in the image based on the discussion in Section 4. We next define a local background area \(N_B(x_k)\) using the “center-surround” approach. For a target area defined by a rectangle of size \(s_x^k \times s_y^k\) another rectangle of size \((2 \times s_x^k) \times (2 \times s_y^k)\) defines the extent of the local background. This is illustrated in Fig.3. It is a common practice to use a weighted histogram for the target’s representation [9, 12]. This is due to the fact that often boundaries between the target and the background are not precise and therefore pixels from the background regions may easily corrupt the histogram of the target area. So a weighting kernel is often used to obtain a weighted histogram by assigning higher trust to pixels closer to the target’s center. The dual foreground-background target model is illustrated in Fig.3.

![Figure 3](image)

Let \(p\) denote the location of the target’s centroid based on the position and size information contained in \(x_k\). The function \(b : \mathbb{R}^2 \to \{1, \ldots, m\}\) maps the intensity value of the pixel at a position given by \(r\), to its bin index in the quantized feature space. The probability of the feature \(u = 1 \ldots m\) is given by [5]

\[
p_{fi}^u(x_k) = \lambda_1 \sum_{r \in N_F(x_k)} K_H(r - p) \delta[b(r) - u],
\]

where \(\lambda_1\) is a normalization constant obtained such that \(\sum_{u=1}^m p_{fi}^u(x_k) = 1\) and \(\delta\) is the Kronecker delta function. We use the triangular kernel for \(K_H\), where the width of the kernel is determined by the target size information contained in \(x_k\). Now we can define a \(m\) dimensional vector \(f_{fi}(x_k) = [p_{fi}^1(x_k), \ldots, p_{fi}^m(x_k)]\) called the foreground histogram. In a similar manner the background area histogram may be obtained as \(f_{bi}(x_k) = [p_{bi}^1(x_k), \ldots, p_{bi}^m(x_k)]\) where

\[
p_{bi}^u(x_k) = \lambda_2 \sum_{p_r \in N_B(x_k)} \delta[b(p_r) - u],
\]

and \(\lambda_2\) is a normalization constant obtained such that \(\sum_{u=1}^m p_{bi}^u(x_k) = 1\). Similarly, we can obtain the foreground histogram \(f_{fs}(x_k)\) and the background histogram \(f_{bs}(x_k)\) from the stdev image \(S_k\). Therefore given \(x_k\), \(I_k\) and \(S_k\) the candidate region is characterized by \(F(x_k)\) defined as follows,

\[
F(x_k) = \{f_{fi}(x_k), f_{bi}(x_k), f_{fs}(x_k), f_{bs}(x_k)\},
\]

which is composed of four different histograms corresponding to each of: the foreground intensity \(f_{fi}(x_k)\), the background intensity \(f_{fs}(x_k)\), the foreground stdev \(f_{fs}(x_k)\), and the background stdev \(f_{bs}(x_k)\). The tracker then evaluates any given candidate area by comparing the similarity of the above four histograms to their known reference models. Thereby we use information from both the foreground and background areas directly in the target tracking process without involving classification.

5.2. Distance measure

During the tracking process we need to evaluate candidate areas based on their distance from a known appearance model of the target denoted by \(F_k\). The reference model \(F_k\) also has a structure similar to \(F(x_k)\) and is given by \(F_k = \{f_{fi,k}, f_{bi,k}, f_{fs,k}, f_{bs,k}\}\). The tracker in essence will compare candidate models \(F(x_k)\) against \(F_k\) based on the histogram intersection (HI) metric first suggested by Swain and Ballard in [11] to measure the similarity between two histograms. The HI metric between any two normalized histograms \(p\) and \(q\) with \(m\) bins each is given by (11)

\[
d(p, q) = \sum_{i=1}^{m} \min(p(i), q(i)).
\]

Every candidate region \(F(x_k)\) comprises four different histograms and therefore we can compute four HI metrics, one corresponding to each pair of histograms in the candidate and the model \(F_k\). Consequently we define our distance metric \(D(F(x_k), F_k; v_{k-1})\) as

\[
D(F(x_k), F_k; v_{k-1}) = \sum_{z \in Z} v_z^{k-1} \cdot d(f_z(x_k), f_z^{k-1}),
\]

where \(Z = \{fi, bi, fi, bs\}\) and \(v_z^{k-1}\) are chosen such that \(\sum_{z \in Z} v_z^{k-1} = 1\). The implication of the \(v_z^{k-1}\) term is to allow for a particular histogram to be more or less dominant in the distance calculation. The values of \(v_z^{k-1}\) are adaptively selected online and this concept is discussed in Section 6 on feature selection. The likelihood \(p(y_k|x_k, v_{k-1})\) is defined based on the distance measure in (12).
5.3. Model update

Since the target is continuously changing with time it is quite intuitive that the reference model $F_k'$ has to be updated to account for this change, otherwise it may result in tracking error. We update our reference model using a simple strategy where past observations are forgotten with time. The reference model $F_{k+1}'$ is obtained based on $\hat{x}_k$, the mean estimate of the states at time step $k$, and the reference model $F_k'$ by using (13) for all $z \in Z$.

$$f_{z,k+1}' = \xi_{z,k} \cdot f_{z,k}' + (1 - \xi_{z,k}) \cdot f_z(\hat{x}_k), \quad (13)$$

where

$$\xi_{z,k} = d(f_z(\hat{x}_k), f'_z), \quad (14)$$

$\xi_{z,k}$ gives the similarity between the model and the estimated histogram. Therefore a sudden change in the target appearance warrants a more aggressive update of the model histograms whereas slower changes do not affect the reference models dramatically.

6. Online feature selection

There are two issues related to online feature selection for target tracking. One is how to evaluate the effectiveness of a certain linear combination of four histogram-based features, and how to evolve the feature weights over time to accommodate the variation of the target appearance. In the following, we will discuss these two issues.

6.1. Feature evolution

The tracker in our model estimates the underlying states based on the distance defined in (12) where the weights $v_{z}^{k-1}$ determines the relative importance of the feature $z$ in the tracking process. A single set of fixed $v_{z}^{k-1}$ may not be the best choice for effective tracking in every sequence. Our idea is that if these weights can be adjusted adaptively based on the sequence at hand, the tracking process will be more robust. So we propose a method in which the $v_{z}^{k-1}$ are updated online in order to accommodate the variability of target’s appearance over time.

We formulate our feature selection in a state space form, where at any time $k$ the state $v_k$ containing the individual feature importance is defined as

$$v_k = [v_{f1}^k, v_{bs}^k, v_{fs}^k, v_{ba}^k], \quad (15)$$

and is subject to the constraint $\sum_{z \in Z} v_{z}^k = 1$. The constraint implies that we have only three independent variables. Therefore it will be easier to decompose $v_k$ into three independent components and then develop a dynamic model for the new transformed variables.

We define a new vector $\Gamma_k = [\alpha_k, \beta_k, \gamma_k]$ such that the individual elements of $v_k$ can be may be expressed as

$$v_{fi}^k = \alpha_k \beta_k,$$

$$v_{bi}^k = (1 - \alpha_k) \gamma_k,$$

$$v_{fs}^k = \alpha_k (1 - \beta_k),$$

$$v_{bs}^k = (1 - \alpha_k)(1 - \gamma_k). \quad (16)$$

Therefore $\Gamma_k$ uniquely determines $v_k$ and is just constrained by the fact $0 \leq \alpha_k, \beta_k, \gamma_k \leq 1$. We then model the dynamics of each component of $\Gamma_k$ as a first order Markov chain with a common predefined step size $\delta$ and equal probability of transition. The dynamics of $\alpha_k$ is given by

$$\alpha_{k+1} = \alpha_k + \epsilon, \quad (17)$$

where $\epsilon \in \{-\delta, 0, \delta\}$ with equal probability. The dynamics of the other parameters ($\beta$ and $\gamma$) can be obtained in a similar way and together they determine the transitional probabilities of the feature weights $p(v_k|v_{k-1})$ that is used to generate possible feature hypotheses for next time step.

6.2. Feature evaluation

Our feature selection is based on the concept that a good feature will result in a higher confidence for the state estimation. Confidence may be described as the measure of how peaky (small variance) the posterior density is at the ground truth state that is not available. Therefore, we assume the present state estimation is accurate based on which we can use the Mahalanobis distance for feature evaluation. Consider the example in Fig. 4, the estimate of the state $x$ is given by the solid line and it has a mean around 0. Now we need to select features that would best estimate $x$. The feature that maximizes the belief of the current state estimation is considered to be the best feature. Based on this idea, all feature hypotheses, i.e., different linear combinations of the four histogram-based distances in 12, are ranked based on the Mahalanobis distance introduced below.

Figure 4. The solid line indicates the current belief of $x$. The other lines correspond to the state estimate using two different features. Feature 2 is considered better than Feature 1 since it shows more confidence on the given state estimation.
Let \( \{x_k^i, w_k^i\}, j = 1 \cdots N_p \) represent the state samples and corresponding weights at time \( k \) and let \( w_{v,k}(j) \) represent the weights for the same set of samples \( x_k^i \) when evaluated with feature \( i \), \( i = 1 \cdots N_v \). Assume all the weights have been normalized. First we compute the weighted mean of the states as \( \bar{x}_k = \sum_{j=1}^{N_p} w_{v,k}(j) x_k^i \). Then we compute the covariance matrix associated with weighted samples as
\[
\Sigma_k = \frac{\sum_{j=1}^{N_p} w_{v,k}(j) (x_k^i - \bar{x}_k)(x_k^i - \bar{x}_k)^\top}{1 - \sum_{j=1}^{N_p} (w_{v,k}(j))^2}. \tag{18}
\]

Now to evaluate each feature we use the information in the weights \( w_{v,k}(j) \). The likelihood of the \( i \)th feature at time \( k \), i.e., \( p(x_k^i | x_k) \), is denoted as \( MD_k^i \) that is defined as
\[
MD_k^i \propto -\sum_{j=1}^{N_p} w_{v,k}(j)(x_k^i - \bar{x}_k)\Sigma_k^{-1}(x_k^i - \bar{x}_k)^\top. \tag{19}
\]

The \( \Sigma_k^{-1} \) term helps us to take into account the variance of the individual state variables and the covariances among them. The \( w_{v,k}(j) \) term in (19) ensures that a feature hypothesis which assigns higher weights to samples closer to the mean will get a better fitness value. Based on the fitness value of every feature hypothesis, we can compute a mean feature value \( \bar{v}_k \) to be used in the next frame.

### 6.3. The complete tracking algorithm

We use two particle filters to obtain the sequential estimation of the target’s states and the feature weights. Once the particle set \( \mathcal{V}_{k-1} \) is available at time \( k-1 \), its mean estimate \( \bar{v}_{k-1} \) is passed on and defines the feature weights at time \( k \). The particle set \( x_k \) corresponding to the tracking state is evaluated based on \( \bar{v}_{k-1} \) to determine the mean state estimate \( \bar{x}_k \). Given the mean estimation, the particle sets of the features \( v_k \) are then updated. The tracking and feature selection are achieved by recursively performing the above process. The pseudo code of the algorithm is provided in Table 1, and the algorithm diagram is shown in Fig. 1.

### 7. Experimental results

The proposed algorithm is evaluated on the AMCOM FLIR dataset. The dataset comprises FLIR sequences in grayscale format (128 x 128 pixels). Information about the target position, size and type are also made available. We perform the tracking on 3 different sequences namely, LW-14-15 from frames 160 through 230, LW-15-NS from frames 160 through 230 and LW-17-01 from frames 1 through 70. We estimate the position and size of the targets at every frame using a particle filtering framework with adaptive feature selection. We present both visual results of tracking and for the first time, quantitative results for the position and size estimation for this dataset.

---

**Table 1.** The pseudo-code of the proposed tracking algorithm.

**7.1. Experimental setup**

Three tracking algorithms were tested that share the same dynamic models described in Section 4. \( \text{PF}_{\text{simple}} \) considers only the foreground intensity histogram for target representation. \( \text{PF}_{\text{dual}} \) extends \( \text{PF}_{\text{simple}} \) by including the proposed dual target model that has equal weights (0.25) for all four histograms. \( \text{PF}_{\text{feature}} \) further extends \( \text{PF}_{\text{dual}} \) by incorporating online feature selection. The target position and size in the first frame of each sequence are initialized with the ground truth.

We set \( N_p = 100 \) for target tracking and \( N_v = 200 \) for feature selection. \( N_v \) determines the number of feature combinations that are evaluated at each step. The computational time of \( \text{PF}_{\text{dual}} \) is about three times as that of \( \text{PF}_{\text{simple}} \), as three additional histograms are used along with the foreground intensity histogram. \( \text{PF}_{\text{feature}} \) has a less than 5% load increase compared with \( \text{PF}_{\text{dual}} \), since it directly uses most of the intermediate results in \( \text{PF}_{\text{dual}} \). In our experiments, we set \( C_k = \beta \text{En}_1(\Delta x_k) \) in (6), \( D = \sqrt{3} \) in (7), the step size for the feature evolution \( \delta \) is set as 0.05 and, the parameter \( \lambda \) used in the evaluation of the weights of the tracking particles (line 7 in Table 1) is set to be 200. The number of bins for the intensity and stdv histograms are set to be 64 and 16 respectively.
Table 2. Mean error of the state variables over 70 frames averaged over 20 Monte Carlo runs using three different algorithms.

<table>
<thead>
<tr>
<th>Sequence</th>
<th>PF&lt;sub&gt;simple&lt;/sub&gt;</th>
<th>PF&lt;sub&gt;dual&lt;/sub&gt;</th>
<th>PF&lt;sub&gt;feature&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x</td>
<td>y</td>
<td>s&lt;sup&gt;x&lt;/sup&gt;</td>
</tr>
<tr>
<td>LW-14-15</td>
<td>3.78</td>
<td>1.86</td>
<td>4.17</td>
</tr>
<tr>
<td>LW-15-NS</td>
<td>1.26</td>
<td>1.04</td>
<td>1.60</td>
</tr>
<tr>
<td>LW-17-01</td>
<td>0.64</td>
<td>1.44</td>
<td>0.99</td>
</tr>
</tbody>
</table>

Figure 5. Partial tracking results (15-20 out of 70 frames) of the state vectors (a) X position (b) Y position (c) X size and (d) Y size for the sequence LW-14-15 using the three different algorithms (Ground truth (––), PF<sub>simple</sub> (–o–), PF<sub>dual</sub> (– -----), PF<sub>feature</sub> (–x–)).

7.2. Results and discussions

The tracking results are shown in Fig. 5 and Table 2 compares the three algorithms on three FLIR sequences. We can observe that PF<sub>feature</sub> and PF<sub>dual</sub> outperform PF<sub>simple</sub> in most cases. Though PF<sub>feature</sub> and PF<sub>dual</sub> perform quite similarly in the position estimation, PF<sub>feature</sub> consistently estimates the size more accurately. It is shown that the use of the dual target model improves the tracking performance, and further improvement is achieved using the feature selection, especially for the sequence LW-14-15. However, in the sequence LW-15-NS the estimation of s<sup>y</sup> has deteriorated with the use of the dual model in comparison to PF<sub>simple</sub>, which is due to the fact that the choice of equal weights for the features may not be optimal for this sequence. This is ascertained by the fact that both the position and size estimation improve in PF<sub>feature</sub>. Also we note that PF<sub>feature</sub> does not show advantages over PF<sub>dual</sub> for the sequence LW-17-01. It is possibly due to the fact that the small size of the target (6×9 pixels) makes the histogram-based feature extraction less robust.

In Fig.6 (a) and (b) we illustrate the sensitivity of the H1 distance to variation in size and position for the different features. We observe that the f<sub>i</sub> is very sensitive to both position and size changes. The f<sub>s</sub> feature is sensitive to position but does not show large variations with change in size. Feature b<sub>s</sub> shows the maximum sensitivity to any size change. Feature b<sub>i</sub> is the least sensitive to position change and is slightly affected by smaller sizes. Fig.6 (c) represents the relative weights of the features during the tracking for the sequence LW-14-15. We observe that feature f<sub>i</sub> is given the most importance followed by b<sub>s</sub>, f<sub>s</sub> and b<sub>i</sub>. This is expected, since features f<sub>i</sub> and b<sub>s</sub> show the maximum sensitivity to the variation of size or position. This result confirms that our feature selection criterion effectively selects the most relevant features for tracking purposes.

8. Conclusions and future work

We have presented a new target tracking algorithm for FLIR imagery that is able to support accurate target localization and size estimation. A dual foreground-background target appearance model integrates local statistics of both background and foreground to enhance the tracker’s sensitivity. Moreover, an online feature selection technique is developed that is able to select optimal features by maximizing the confidence of the state estimation for target tracking. The proposed method differs from many state-of-the-art tracking algorithms where online feature selection aims at maximizing the discriminability between foreground and background. In this work, both target tracking and feature selection are unified in a probabilistic framework where a coupled particle filtering approach is involved for sequential state estimation. Tracking results on real FLIR imagery show the improved performance of the dual model over the foreground-based tracker. The use of online feature selection further improves the tracking performance.

One limitation is that the online feature selection does not show improvement when the target is very small (e.g., less than 50 pixels in the target area). We think it is because that the small targets make histogram-based feature extraction unreliable due to the fixed bin numbers and thus reduce the merit of feature selection. A possible remedy is to adaptively adjust bin numbers according to the target size. It is expected that the proposed approach is applicable to the cases involving multiple sensors or feature domains.

Acknowledgements

The authors would like to acknowledge the anonymous reviewers for their constructive comments that have improved this paper. This work is supported by the Army Research Office (ARO) under Grant W911NF-04-1-0221.
Figure 6. Variations of the HI distance defined in (11) with respect to the change in positions (a) and sizes (b) for four different features. (c) Variations of the relative importance between four different features during the tracking process of the FLIR sequence LW-14-15.

Figure 7. The tracking gates produced by algorithm PF_{feature} for sequences LW-15-NS (top row) and sequence LW-14-15 (bottom row).

References