Reducing Processing Time for Histogram PMHT Algorithm in Video Object Tracking

Ahmet Güngör Pakfiliz (iD)

Department of Electrical and Electronics Engineering, Başkent University School of Engineering, Ankara, Turkey

ABSTRACT

This paper describes a novel approach for reducing the processing time of the histogram probabilistic multi-hypothesis tracker (H-PMHT) algorithm in video applications. Video data of flying vehicles is taken from surface to air, and a temporal difference-based technique is applied to video frames for meeting the intensity demands of H-PMHT algorithm. This technique also enables discrimination between objects and eliminates clutter. Variations between the structures of the standard and the improved version of H-PMHT algorithms are described. In addition, the improved H-PMPT is compared with the standard H-PMHT and another approved tracking algorithm to evaluate the performance and processing time reduction ratings.

Keywords: Improved H-PMHT, Pixel Wise Difference, Surface to air, Video Tracking

Introduction

TRACKING requires high precision and real-time applications. For sensors taking continuous and bulky data streams such as video data, real-time operation is difficult to achieve using whole sensor data. When conventional tracking algorithms are used for video object tracking real-time may be achieved, but only at the expense of some level of precision due to transformation of the data into a suitable measurement domain [1-3]. These transformations also take the physical shapes of the targets, which are projected onto the image frame, up to a level and convert them to point measurement, thus adding to measurement error. Coping with a high amount of data streams is necessary in order to reach high precision rates in video object tracking. Therefore, target shape and intensity-based algorithms get the edge on point-measurement trackers [4, 5]. Reliable and uninterrupted track information is essential for most of the applications, especially in a background of a high clutter environment and a lack of sensor capacity. In order to detect the target location from the data stream, high intensity pixel clusters should be searched from data streams, and it should be decided whether or not they emerge from target or clutter. H-PMHT is a feasible method for handling data streams, and for tracking objects reliably and uninterruptedly.

The H-PMHT is basically an Expectation Maximization (EM) based algorithm which was developed for target tracking in dense clutter environment by processing a considerable amount of data streams [6]. H-PMHT is a track-before-detect (TkBD) algorithm and entire video data is used as the measurement. It processes detection and tracking operations simultaneously. H-PMHT maintains tracking performance for low SNR values, where the target is not easily distinguished from the noise-cluttered background of any given frame. The original H-PMHT assumes that the signature of each target in the sensor frame area is known. In this particular case, the signatures are in the Gaussian distribution, and the means of these Gaussians are linearly related to the states of the targets. In imagery applications, the target signature is the physical shape of the target projected onto the image frame. This shape can be time-varying and complicated [7]. H-PMHT provides considerably satisfactory results for one dimensional and two dimensional applications [8-10]. In these applications spreading of the target inten-

sities presents almost a linear-Gaussian distribution. Moreover, for non-linear and non-Gaussian applications a particular solution is presented in with particle filters. In addition, a video tracking application is presented in with a specially processed video data, and a modified H-PMHT, which is called H-PMHT with Random Matrices (H-PMHT-RM) [11, 12].

The main purpose of this study is to reduce the processing time of the H-PMHT algorithm without deterioration in performance. For this purpose, we needed to obtain available and reliable measurement data in order to discriminate moving objects and represent them with a higher intensity area than the stationary background. Firstly, video data belonging to air vehicles is taken in true color (RGB) from surface to ground. Then a temporal difference processing technique is applied to filter moving objects from the stationary background. Thereby, proper data streams composed of moving objects and background are obtained for processing with H-PMHT.

Two important obstacles need to be overcome to reduce the processing time of H-PMHT. One of them is the dynamic structure of the video data, because in video tracking applications tracker processes data sequentially, not in a batch structure. The other is the long processing time, which is due to the structure of H-PMHT. By setting the batch number to one and replacing the smoother filter by a Kalman filter it becomes possible to overcome the issue of data processing sequence. On the other hand, dealing with the long processing time is a challenge. The aim of this study is not to completely eliminate this issue, but to reduce the processing time by improving H-PMHT algorithm. For this purpose, not only algorithm improvement, but also an amendment needs to take place in the basic structure of the H-PMHT in order to reduce the processing time while continuing tracking and keeping estimation error within a reasonable limit. The resulting algorithm is called Improved H-PMHT (I/H-PMHT).

The tracking performance of I/H-PMHT for video data is given in the experimental study section. In this section, the obtained results are also compared in terms of processing time and estimation error with standard H-PMHT, and Interacting Multi Model Probabilistic Data Association with Amplitude Information (IM-MPDA-AI) algorithms for different conditions and cases [13].

Measurement Model

This study intends to reduce the processing time of H-PMHT algorithm. To that end, RGB video is taken for various aircrafts and each frame is processed separately according to time sequence. First of all, RGB images are converted into intensity images. National Television System Committee (NTSC) standard for transforming RGB to grayscale, defined in, is given as follows,

$$I(x,y) = 0.2989 R(x,y) + 0.587 G(x,y) + 0.114 B(x,y) \quad (1)$$

After that, pixel-wise difference function is obtained using a similar process as that described in [14, 15]. The aim of this process is to obtain frame data for frames k and $k+\Delta$, then subtract them and take the absolute value of the difference. This is followed by the thresholding process, and the remaining values give information about the movement. k represents the instantaneous frame time, and Δ represents the sampling period of frames. As long as the object does not move too fast, the detected regions of the object are reduced. However, this process leaves a ghost where the object was located, and a large part of the object cannot be detected unless the object moves fast [16]. Additional processing is required to reduce the ghost effect by adapting the measurement data to the situation. For this purpose, we obtained the difference from the former to the following time sequence as follows.

$$I_{dif}(x,y) = I_k(x,y) - I_{k+\Delta}(x,y) \quad (2)$$

There are two additional steps that must be performed to obtain the measurement model. The first step is thresholding the unwanted ghosts due to the former time k. As a result of the operation in equation (2), negative and positive intensity differences are obtained. Indeed, high magnitude negative values are out of the scope of this work, because they are the reflections of former target echo. In order to get rid of these spurious intensities, negative values are converged to noise floor, and positive values are taken directly. Thus the intensity level of each cell $M(x,y)$ is given as follows,

$$M(x,y) = \begin{cases} I_{Noise\_floor} + rand & , I_{dif}(x,y) \le 0 \\ I_{dif}(x,y) = |I_{dif}(x,y)| & , I_{dif}(x,y) > 0 \end{cases} \quad (3)$$

where I_{Noise_floor} represents the intensity of the noise floor, which is defined as the intensity level of pixels that do not originate from the target or the clutter. $I_{dif}(x,y)$ represents the intensity of differences, and $rand$ is the uniformly distributed pseudorandom number.

The second step taken in order to obtain the measurement model is intensity pruning. The square root of $M(x,y)$ values of intensities are taken as in equation (4). Thus, an excessive increment in dynamic range and high intensity clutters are prevented.

$$I_{PDIM}(x,y) = \sqrt{M(x,y)} \quad (4)$$

The measurement model is achieved after obtaining pixel intensity levels for each pixel of the sensor area. Because the data is obtained using the difference of pixel intensities of the sequential video frames, it is defined as Pixel-wise Difference Intensity Modulated (PDIM) data. RGB images of sequential video frames of an aircraft, intensity images related to them, and resulting PDIM data image are given in Figure 1. The scan steps between them are 3 frames, in other words $\Delta=3$.

Using PDIM method, stationary parts of the image can be eliminated or reduced to acceptable levels, since intensity levels of
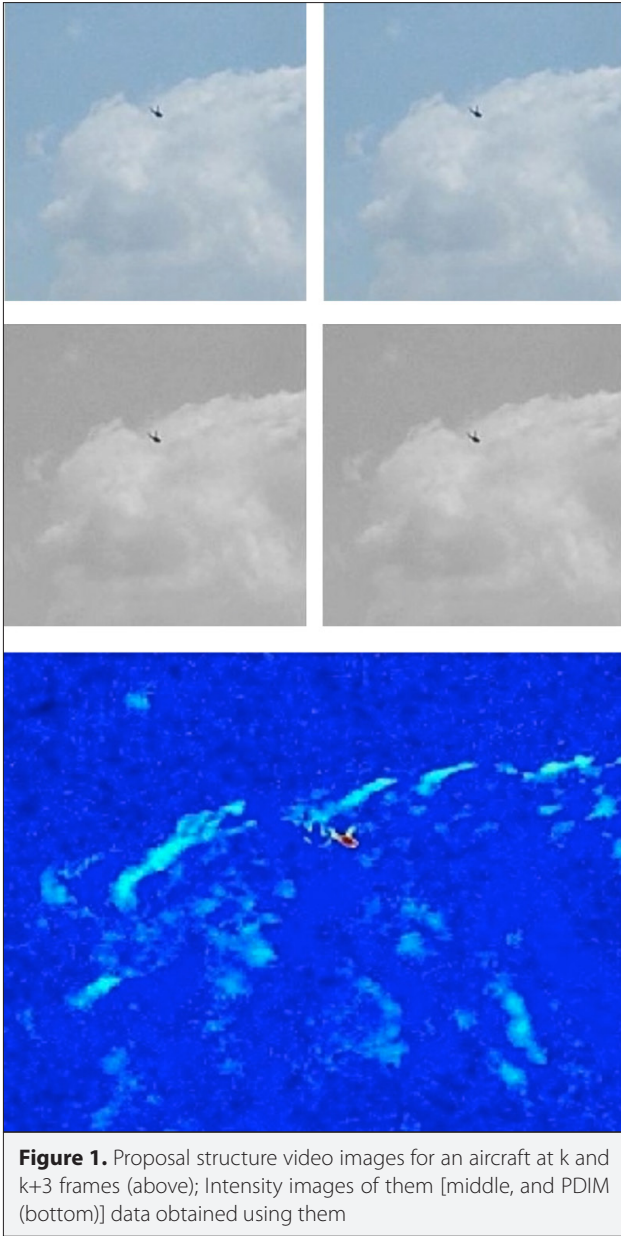
Figure 1. Proposal structure video images for an aircraft at k and k+3 frames (above); Intensity images of them [middle, and PDIM (bottom)] data obtained using them

pixels related to stationary parts will remain the same at consecutive times and the difference of the intensity levels will be zero, small motions on some parts of stationary objects or a slight vibration of camera will result in a non-zero difference level, and it is taken into calculations as background clutter. In this study PDIM data is used for surface to air video object tracking with improved H-PMHT algorithm.

Basic Structure of H-PMHT

The H-PMHT algorithm is introduced in [6, 8, 9] with its theory and derivations. Before entering the details of H-PMHT algorithm, its parametric TkBD structure is discussed. In traditional tracking methods, thresholding, clustering, extracting and

tracking procedures are carried out consecutively. On the other hand, TkBD method performs all the steps concurrently [17, 18]. TkBD merges detection and estimation phases by eliminating the detection algorithm from the process and supplying the whole sensor frame directly to the tracker. This increases trace accuracy and allows the tracker to keep track for low SNR targets [19]. The H-PMHT algorithm inherently includes the TkBD capability and makes it possible to obtain extended object traces directly from an image sequence.

Only a general structural outline for H-PMHT is given here. H-PMHT is mainly developed from PMHT, and all derivations of PMHT arise from Expectation Maximization (EM) method [20]. The purpose of using the EM process in the H-PMHT algorithm is to assign histogram distribution to the model components and to designate the precise position of the shots as missing data. It also allows for unobserved cells with abstract sensor pixels that do not convey any data. The probability of the missing data is determined in the E-step and the state estimates are refined in the M-step. Initialization and iteration steps of H-PMHT according to E and M-steps are given below.

Initialization of H-PMHT Algorithm

Initialization steps must be defined before iterations are described. At the beginning of each iteration mixing proportions ($\hat{\pi}_{tk}^{(0)}$ are determined for background and all target models (k=0, 1,…, M), and for batch sequence $t = 1 = 1, K, T$, for which T≥1 denotes the number of scans in a batch of measurement, as follows,

$$\hat{\pi}_{tk}^{(0)} > 0 \quad and \quad \hat{\pi}_{t0}^{(0)} + \hat{\pi}_{t1}^{(0)} + \cdots + \hat{\pi}_{tM}^{(0)} = 1 \tag{5}$$

In addition, for $k = 1, .K, M$, for which M≥1 denotes the number of targets, the following is initialized,

- Target State Sequence: $\hat{x}_{0k}^{(0)}, \hat{x}_{1k}^{(0)}, \ldots, \hat{x}_{Tk}^{(0)}$

- Measurement Covariance Sequence: $\hat{R}_{1k}^{(0)}, \ldots, \hat{R}_{Tk}^{(0)}$

- Target Covariance: $\hat{Q}_{k}^{(0)}$

The H-PMHT algorithm consists of repeated iteration steps for each batch sequence $t = 1, \ldots, T$. Some of these iteration steps stem from Expectation and Maximization, the remainder comes from Kalman smoothing filter. Throughout the iterations the dynamic matrix F and measurement matrix H are assumed as constant or time invariant. Iteration steps with respect to Expectation, Maximization, and Kalman smoothing processes are given in the following subsections.

Expectation

Step 1. Total Sensor Probabilities (TSPs)

First, Target Cell Probabilities (TCP) are calculated for batch length t=1,…,T for all cells ℓ=1,…,S and for all target models, including background k=0,1,…,M . S represents the number of

whole cells in the sensor area. For the background and targets TCPs are calculated as follows,

$$
P_{tk\ell}^{(i+1)} = \begin{cases} \dfrac{1}{S} & if \ k = 0 \\ \displaystyle\int_{B_\ell(t)} N\left(\tau; H_{tk}\hat{x}_{tk}^{(i)}, \hat{R}_{tk}^{(i)}\right) d\tau & if \ k = 1,\dots,M \end{cases}
\tag{6}
$$

where τ represents the variable term of $N(\cdot)$ Gaussian PDF.

Total cell probabilities are obtained by summing the product of TCPs and mixing proportions for all target models and background as in equation (7).

$$
P_{t\ell}^{(i+1)} = \sum_{k=0}^{M} \hat{\pi}_{tk}^{(i)} P_{tk\ell}^{(i+1)}
\tag{7}
$$

Finally, TSPs are attained using only displayed cells or cells with measurement $\ell=1,\dots,L(t)$ as follows;

$$
P_{t}^{(i+1)} = \sum_{\ell=1}^{L(t)} P_{t\ell}^{(i+1)}
\tag{8}
$$

Step 2. Expected Measurements (EMs)

EMs are calculated as in equation (9) for $t=1,\dots,T$, and $\ell=1,\dots,S$.

$$
\overline{z}_{t\ell}^{(i+1)} = \begin{cases} z_{t\ell} & 1 \le \ell \le L(t) \\ \left\|Z_t\right\| \left(\dfrac{P_{t\ell}^{(i+1)}}{P_{t}^{(i+1)}}\right) & L(t)+1 \le \ell \le S \end{cases}
\tag{9}
$$

where $\left\|Z_t\right\|$ represents L1 norm of displayed cells $\left\{B_1(t),\dots,B_{L(t)}(t)\right\}$ and it is defined as follows,

$$
\left\|Z_t\right\| = \sum_{\ell=1}^{L(t)} z_{t\ell}
\tag{10}
$$

Maximization

Step 3. Cell-level Centroids (CCs):

CCs are calculated by using equation (11).

$$
\tilde{z}_{tk\ell}^{(i+1)} = \frac{1}{P_{tk\ell}^{(i+1)}} \int_{B_\ell(t)} \tau N\left(\tau; H_{tk}\hat{x}_{tk}^{(i)}, \hat{R}_{tk}^{(i)}\right) d\tau
\tag{11}
$$

Using CCs, synthetic measurements are obtained as follows;

$$
\tilde{z}_{tk}^{(i+1)} = \frac{\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]\tilde{z}_{tk\ell}^{(i+1)}}{\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]} \rightarrow \left\{\wp_{tk\ell} = \frac{P_{tk\ell}^{(i+1)}}{P_{t\ell}^{(i+1)}}\right\}
\tag{12}
$$

Step 4. Synthetic Covariance Matrices

Synthetic measurement matrices given in equation (13) are obtained for $t = 1,\dots,T$, and $k = 1,\dots,M$.

$$
\tilde{R}_{tk}^{(i+1)} = \frac{\hat{R}_{tk}^{(i)}}{\hat{\pi}_{tk}^{(i)}\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]}
\tag{13}
$$

Also, for $t = 0,1,\dots,T-1$ synthetic measurement covariance matrices are calculated as follows:

$$
\tilde{Q}_{tk}^{(i+1)} = \frac{P_{t+1}^{(i+1)}}{\left\|Z_{t+1}\right\|}\hat{Q}_{k}^{(i)}
\tag{14}
$$

Step 5. Mixing Proportions

Mixing proportions are calculated for $t = 1,\dots,T$ and $k = 0,1,\dots,M$:

$$
\hat{\pi}_{tk}^{(i+1)} = \frac{\hat{\pi}_{tk}^{(i)}\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]}{\sum_{k'=0}^{M}\hat{\pi}_{tk'}^{(i)}\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]}
\tag{15}
$$

Kalman Smoothing Filter

To obtain estimated target states a Kalman smoother filter is applied. This portion of the algorithm is composed of forward and backward filters.

Step 6. Forward Filter

The forward Kalman smoother filter for $t=0, 1,\dots, T-1$ is applied using synthetic measurements in order to refine target state estimates. At this point dummy expectation is taken as $\tilde{y}_{0|0}^{(i+1)}(k) = 0$ and dummy covariance is $P_{0|0}^{(i+1)}(k) = 0$. The equations of forward filter are given in (16)-(18)

$$
P_{t+1|t}^{(i+1)}(k) = F P_{t|t}^{(i+1)}(k) F^{*} + \tilde{Q}_{tk}^{(i+1)}
\tag{16}
$$

$$
W_{t+1}^{(i+1)}(k) = P_{t+1|t}^{(i+1)}(k) H \left(H\ P_{t+1|t}^{(i+1)}(k) H^{*} + R_{t+1,k}^{(i+1)}\right)^{-1}
\tag{17}
$$

$$
P_{t+1|t+1}^{(i+1)}(k) = P_{t+1|t}^{(i+1)}(k) - W_{t+1}^{(i+1)}(k) H\ P_{t+1|t}^{(i+1)}(k)
\tag{18}
$$

$$
\tilde{y}_{t+1|t+1}^{(i+1)}(k) = F\tilde{y}_{t|t}^{(i+1)}(k) + W_{t+1}^{(i+1)}(k)\left(\tilde{z}_{t+1,k}^{(i+1)} - H\tilde{y}_{t|t}^{(i+1)}(k)\right)
\tag{19}
$$

where $W_{t+1}^{(i+1)}$ is filter gain, and F is state transition matrix, which are defined in [4, 5].

Step 7. Backward Filter

The equation of backward filter for $t = T - 1, ... 1$ is given as follows:

$$\hat{x}_{tk}^{(i+1)} = \tilde{y}_{t|t}^{(i+1)}(k) + P_{t|t}^{(i+1)}(k) F^* \left(P_{t+1|t}^{(i+1)}(k) \right)^{-1} (ln)$$

$$\text{where } ln = \hat{x}_{t+1,k}^{(i+1)} - F\tilde{y}_{t|t}^{(i+1)}(k) \tag{20}$$

Step 8. Estimated Covariance Matrices

First cell-level measurement covariance is calculated,

$$\hat{R}_{tk\ell}^{(i+1)} = \frac{\int_{B_\ell(t)} N\left(\tau; H_{tk}\hat{x}_{tk}^{(i)}, \hat{R}_{tk}^{(i)}\right) E\ E^*\ d\tau}{P_{tk\ell}^{(i+1)}} \tag{21}$$

$$\text{where } E = \tau - H_{tk}\hat{x}_{tk}^{(i+1)}$$

The estimated measurement covariance matrix is calculated as given in equation (22):

$$\hat{R}_{tk}^{(i+1)} = \frac{\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]\hat{R}_{tk\ell}^{(i+1)}}{\sum_{\ell=1}^{S}\left[\overline{z}_{t\ell}^{(i+1)}\left(\wp_{tk\ell}\right)\right]} \tag{22}$$

And the last operation shown in equation (23) of the iteration is to obtain estimated target covariance matrices for all the target models except for the background.

$$\hat{Q}_k^{(i+1)} = \frac{\sum_{t=1}^{T}\left(\frac{\|Z_t\|}{P_t^{(i+1)}}\right)\left(\hat{x}_{tk}^{(i+1)} - F\hat{x}_{t-1,k}^{(i+1)}\right)\left(\hat{x}_{tk}^{(i+1)} - F\hat{x}_{t-1,k}^{(i+1)}\right)^*}{\sum_{t=1}^{T}\left(\frac{\|Z_t\|}{P_t^{(i+1)}}\right)} \tag{23}$$

Variations of the Algorithm

Histogram probabilistic multi-hypothesis tracker is a strong and reliable tracking algorithm and originally developed for data streams. However, it is not conformed to real time applications. The aim of this study is to converge real time tracking or reduce the processing time of H-PMHT algorithm without terminating tracking for video applications. First of all, video data is converted to PDIM data in order to clutter effects and enhance target detectability. Then some improvements are applied to mathematical operations, in particular by taking the practical advantage of two-dimension, and also some amendments are applied to the algorithm itself. This work was done by adding intensity information that can be thought of as a third dimension in two-dimensional space. For this reason, this study is regarded as a two-dimensional application and the assumptions in [10] can be implemented. The most important aspect of these assumptions is that x and y axes are statisti-

cally independent of each other. Another assumption for the process is that there is no pre-information about point spread function of the objects. They mostly retain their original shape and this shape is not similar to linear-Gaussian distribution.

It will be proper to state that there is no revision on the sequence of EM based iteration steps. The main difference takes place in the batch structure of EM iteration. In this study batch structure is eliminated and its structure is converted to single scan algorithm. To overcome a priori information absence, a priori density obtained via earlier measured data is used as stated in [6]. Using single scan structure is more viable than batch structure for converging real-time video object tracking applications. By using single scan structure, smoothing filter turns to Kalman Filter (KF), but in this case not much processing time reduction takes place. This is the first step for the reduction of processing time. The improvements and amendments are described separately in the following subsections.

Operational Improvements

In this section, no algorithmic amendments, but operation improvements of the algorithm are described. To achieve this aim the batch structure is turned into single scan algorithm, but multi iteration structure is preserved. The inspected part in this section is the most time-consuming fragment of the H-PMHT process which is spent in calculation of integration operations. These operations are

- Target Cell Probabilities $P_{tk\ell}^{(i+1)}$,

- Cell Level Centroids $\overline{z}_{tk\ell}^{(i+1)}$,

- Cell Level Meas. Covariance Contributions $\hat{R}_{tk\ell}^{(i+1)}$.

In two-dimensional case these three expressions are normally calculated for x and y axes separately, then corresponding values of each pixel are multiplied with each other to obtain the overall value. The total number of integration (NoI) is calculated by summing up NoI of the above three expressions. In this context NoI of a sensor area with "200 x 200 pixels" is calculated as follows.

For x-axis:

$$NoI\left\{P_{tk\ell_x}^{(i+1)}\right\} = NoI\left\{\tilde{z}_{tk\ell_x}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell_x}^{(i+1)}\right\} = 40000$$

For y-axis:

$$NoI\left\{P_{tk\ell_y}^{(i+1)}\right\} = NoI\left\{\tilde{z}_{tk\ell_y}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell_y}^{(i+1)}\right\} = 40000$$

For two dimensions:

$$NoI\left\{P_{tk\ell}^{(i+1)}\right\} = NoI\left\{\tilde{z}_{tk\ell}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell}^{(i+1)}\right\} = 2 \times 40000$$

$$= 80000$$

Total NoI:

$$Total - NoI = 6 \times 80000 = 480000$$

This phase increases the process time exponentially for linear increasing of sensor dimensions. To decrease the processing time, an improvement method is applied to H-PMHT operation in order to obtain Target Cell Probabilities, Cell Level Centroids, and Cell Level Measurement Covariance Contributions. For this purpose, only the elements of the first row of x axis contribution and only the elements of the first column of y axis contribution are calculated. Thus, NoI for sensor area with "200 x 200" pixels can be shown as follows.

For x-axis:

$$NoI\left\{P_{tk\ell_x}^{(i+1)}\right\} = NoI\left\{\bar{z}_{tk\ell_x}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell_x}^{(i+1)}\right\} = \begin{array}{c} No.of \ Row \\ \left(NoR\right) \end{array}$$

For y-axis:

$$NoI\left\{P_{tk\ell_y}^{(i+1)}\right\} = NoI\left\{\bar{z}_{tk\ell_y}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell_y}^{(i+1)}\right\} = \begin{array}{c} No.of \ Column \\ \left(NoC\right) \end{array}$$

For two dimensions:

$$NoI\left\{P_{tk\ell}^{(i+1)}\right\} = NoI\left\{\bar{z}_{tk\ell}^{(i+1)}\right\} = NoI\left\{\hat{R}_{tk\ell}^{(i+1)}\right\} = NoR + NoC$$
$$= 2 \times 200 = 400$$

Total NoI:

$$Total - NoI = 6 \times pixel - no. = 6 \times 400 = 2400$$

After finding the elements of the first row of x axis contribution for the three expressions, we then selected the other rows in the same way as the first row and established Target Cell Probabilities, Cell Level Centroids, and Cell Level Measurement Covariance Contributions. Similarly, for y axis contribution, other columns are taken in the same way as the first column. In fact, the results obtained by taking integration for whole pixels of the sensor area using the classical method is the same as the reduced one. Thus, the NoI reduces from 480000 to 2400. A remarkable reduction in processing time is obtained and no reduction in accuracy occurs Applying this improvement approximately 34 - 63% (differs for different sensor area) process time reducing with respect to standard H-PMHT is obtained. The results are given in the experimental study section.

Algorithm Amendments

In addition, some algorithm amendments were used in order to decrease processing time and adjust the algorithm structure to dynamic and real time conditions. These amendments are given in the following items.

Item 1. First of all, the batch structure of the H-PMHT algorithm is converted to single scan algorithm, and the algorithm becomes more suitable for dynamic applications, such as video object tracking [19]. A result of this conversion that backward filter is removed and the smoothing filter turns into KF. Because smoother filer turns to KF, the estimated target states $\tilde{y}_{t|t}^{(i+1)}(k)$ and dummy covariance matrix $P_{t|t}^{(i+1)}(k)$ are picked up from the previous frame step instead of initiate them for each itera-

tion. Also, the initiation process is eliminated and the output of the previous time will be the input of the next time.

Item 2. Reducing iteration number is another amendment for reducing processing time. But reducing iteration number without taking necessary measures may be resulted with convergence insufficiency of state estimates to true position. In order to mitigate this risk, measurement covariance sequences in the initialization phase $\hat{R}_{tk}^{(0)}$ are selected compatible with the targets. Compatible means selecting initial measurement covariance low for small targets, and high for large targets.

Item 3. In order to reduce iteration number without increasing the estimation error more than an allowable amount, an additional measure is taken. In the fundamental theory of H-PMHT [6], some expressions are calculated for only displayed cells $(B_1(t),...,B_{L(t)}(t))$, without using the truncated ones $(B_{L(t)+1}(t),...,B_S(t))$. These expressions are; L1 norm of measurements $\|Z_t\|$, total sensor probabilities $P_t^{(i+1)}$, and expected measurements $\bar{z}_{t\ell}^{(i+1)}$. The border level between displayed and truncated cells can be regarded as threshold level which comes from the structure of H-PMHT. In fact, this is not exactly a threshold, because truncated cells are also counted in the calculations except the above expressions. Thus it may be called as *Displayed Cell Threshold*-DCT. Selecting a higher DCT value will reduce the number of steps required to converge the state estimates to the correct position. On the other hand, increasing the DCT excessively may cause some target-based measurements to be incomplete.

Item 4. Using compatible initial measurement covariance $\hat{R}_{tk}^{(0)}$ and proper DCT, the iteration number may be reduced to one, and an additional reduction in processing time can be reached. Calculation of estimated measurement matrix $\hat{R}_{tk}^{(i+1)}$ is not necessary for one iteration case.

-By using compatible initial measurement covariance $\hat{R}_{tk}^{(0)}$ and proper DCT, iteration number may be reduced to one, and an additional reduction in processing time can be reached. Calculation of estimated measurement matrix $\hat{R}_{tk}^{(i+1)}$ is not necessary for one iteration case.

After making these amendments approximately 8 -29% additional process time reducing (differs for different sensor area) is obtained. This reduction takes place after operational improvement, and the reduction rate is based on the CPU time of operational improvement applied H-PMHT, not the total CPU time rate. This case is different from operational improvements, because algorithm amendments result in sacrificing some accuracy, especially in low DCTs.

At the end of the improvement and amendment process the total decreasing process time converges to approximately 65%. The reduction amount decreases with big sensor area, and it increases with small sensor area. The detailed results are also given in the simulation section. The new version of the algorithm is named as improved H-PMHT (I/H-PMHT), and schematic structure is given in Table 1.

Table 1. Structural Comparison between Standard H-PMHT and I/H-PMHT

Phase of the Algorithms	Standard H-PMHT	I/H-PMHT Multi Iteration (Only Operational Improvement)	I/H-PMHT One Iteration (Both Operational Improvement and Algorithmic Amendment)
Initialization	for batch length $t = 1,\ldots T$ $$\hat{x}_{0k}^{(0)}, \hat{x}_{1k}^{(0)}, \ldots, \hat{x}_{Tk}^{(0)}$$ $$\hat{\pi}_{t0}^{(0)} + \hat{\pi}_{t1}^{(0)} + \cdots + \hat{\pi}_{tM}^{(0)}$$ $$\hat{R}_{1k}^{(0)}, \ldots, \hat{R}_{Tk}^{(0)} \quad and \quad \hat{Q}_k^{(0)}$$	no batch for each scan $$\hat{x}_{tk}^{(0)}, \hat{\pi}_{tk}^{(0)}$$ $$\hat{R}_{tk}^{(0)} \text{ and } \hat{Q}_k^{(0)}$$	no batch for each scan $$\hat{x}_{tk}, \hat{\pi}_{tk}$$ $$\hat{R}_{tk}^{(0)} \text{ and } \hat{Q}_k^{(0)}$$
Expectation	$$P_{tk\ell}^{(i+1)}, P_{t\ell}^{(i+1)}$$ $$P_t^{(i+1)}, \overline{z}_{t\ell}^{(i+1)}$$	$P_{tk\ell}^{(i+1)}$ (Operational impr.) $$P_{t\ell}^{(i+1)}, P_t^{(i+1)}, \overline{z}_{t\ell}^{(i+1)}$$	$P_{tk\ell}^{(i+1)}$ (Operational impr.) $$P_{t\ell}^{(i+1)}, P_t^{(i+1)}, \overline{z}_{t\ell}^{(i+1)}$$
Maximization	$$\tilde{z}_{tk\ell}^{(i+1)}, \overline{z}_{tk}^{(i+1)},$$ $$\tilde{R}_{tk}^{(i+1)}, \tilde{Q}_{tk}^{(i+1)}, \hat{\pi}_{tk}^{(i+1)}$$	$\tilde{z}_{tk\ell}^{(i+1)}$ (Operational impr.) $$\overline{z}_{tk}^{(i+1)}, \tilde{R}_{tk}^{(i+1)}$$ $$\tilde{Q}_{tk}^{(i+1)}, \hat{\pi}_{tk}^{(i+1)}$$	$\tilde{z}_{tk\ell}^{(i+1)}$ (Operational impr.) $$\overline{z}_{tk}^{(i+1)}, \tilde{R}_{tk}^{(i+1)}$$ $$\tilde{Q}_{tk}^{(i+1)}, \hat{\pi}_{tk}^{(i+1)}$$
Kalman Smoothing Filter	Forward Filter $$\tilde{y}_{t+1|t+1}^{(i+1)}(k)$$ Backward Filter $$\hat{x}_{tk}^{(i+1)}$$	Forward Filter $$\hat{x}_{tk}^{(i+1)}$$ Backward Filter (No Backward Filter)	Forward Filter $$\hat{x}_{tk}^{(i+1)}$$ Backward Filter (No Backward Filter)
Covariance Estimation	$$\hat{R}_{tk\ell}^{(i+1)}$$ $$\hat{R}_{tk}^{(i+1)}$$ $$\hat{Q}_{tk}^{(i+1)}$$	$\hat{R}_{tk\ell}^{(i+1)}$ (Operational impr.) $$\hat{R}_{tk}^{(i+1)}$$ $$\hat{Q}_{tk}^{(i+1)}$$	$\hat{R}_{tk\ell}^{(i+1)}$ (not necessary) $\hat{R}_{tk}^{(i+1)}$ (not necessary) $$\hat{Q}_{tk}^{(i+1)}$$

Experimental Study

The experimental trials were conducted for different scenarios of single aircraft and chopper videos taken from surface to air. Video data was captured in true color (RGB) format with "640x480" pixels, using a 14.1 Mega-pixel camera. In the simulations the dimensions of sensor areas are selected "200x200", "250x250", and "300x300" pixels in order to evaluate CPU reduction rate for different sensor areas.

The study is conducted in two-dimensional case, and the assumptions made in [10] are used. Additionally, pre-information about point spread function of the objects does not exist. They mostly retain their original shape and these shapes are quite different from the linear-Gaussian distribution.

To perform the operation, first the PDIM data is obtained using the video data taken from surface to air in daylight. Then tracking process is applied to each PDIM data using I/H-PMHT algorithm. For benchmarking the obtained results in terms of process time reduction and performance each scenario is re-applied to standard H-PMHT and a trustworthy probabilistic algorithm IMMPDA-AI. This algorithm is a combination of IMM estimator and PDA technique [4, 5], and by adding amplitude information, the results obtained with IMMPDA-AI will be proper for a fair comparison with the results of I/H-PMHT. Thus, processing time shortening and tracking performances of I/H-PMHT are compared with the counterparts of standard H-PMHT and IMMPDA-AI.

Interacting Multi Model Probabilistic Data Association with Amplitude Information is a point tracker method. PDIM data was adapted to meet the requirements without violating fair comparison. To obtain measurement data suitable with IMMPDA-AI, first the mean value of total sensor area (say, nxn) intensity ratio ($\overline{I}_{SA}$) is calculated. Threshold is selected 10% higher than the mean value, and the resulting value is taken as amplitude information threshold (T_{AI}). Hence the number of point measurements is brought into a reasonable level for comparison. Then the whole the sensor area is divided into 5x5 pixel units and mean intensity values of each pixel unit ($\overline{I}_{PU}$) are calculated as follows,

$$\overline{I}_{SA} = \frac{\sum_{all-pxl} I(x,y)}{n \times n} \qquad T_{AI} = 1.1 \times \overline{I}_{SA}$$

$$\overline{I}_{PU(5\times5)} = \frac{\sum_{5\times5} I(x,y)}{5 \times 5} \tag{24}$$

The mean intensity values of each pixel unit compare with the threshold magnitude. If the mean intensity of a pixel unit is greater than the threshold of amplitude information $\overline{I}_{PU} \geq T_{AI}$ then it is accepted as a measurement, otherwise it is not taken as a measurement. For each measurement, the mean intensity of the related pixel unit ($\overline{I}_{PU}$) is taken as amplitude information.

In the experimental study the target parameters of interest are the location and velocity according to *x* and *y* axes, and the state vector is as follows,

$$X_{tk} = \begin{bmatrix} \mathbf{x_{tk}} & \left(\dot{\mathbf{x}}_{tk}\right) & \mathbf{y_{tk}} & \left(\dot{\mathbf{y}}_{tk}\right) \end{bmatrix}^{\mathbf{T}} \tag{25}$$

at time *t* and for target *k*. *k = 1* is selected because single target is assumed. For the obtained four-state Markov model, state transition and process covariance matrices are as follows:

$$F = \begin{bmatrix} 1 & \Delta & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad Q_k = \sigma^2 \begin{bmatrix} \Delta^3/3 & \Delta^2/2 & 0 & 0 \\ \Delta^2/2 & \Delta & 0 & 0 \\ 0 & 0 & \Delta^3/3 & \Delta^2/2 \\ 0 & 0 & \Delta^2/2 & \Delta \end{bmatrix} \tag{26}$$

where Δ represents the frame numbers between samples of video data, and σ is the process noise standard deviation or scale factor as defined in [21]. Measurement matrix *H*, and measurement covariance matrix R_{tk} are defined as follows:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad R_{tk} = \rho^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{27}$$

where ρ^2 is measurement error variance.

The performances of I/H-PMHT, standard H-PMHT and IMMPDA-AI were analyzed using real-life records surface to air video data with single chopper or aircraft. Various scenarios were employed to assess the performances of algorithms in different environments, speeds and target geometries. The environmental conditions directly affect the signal-to-noise ratio (SNR), so different SNR values are used in the study. In addition to different aircraft types, different speed ratios, different pixel propagation levels and deviations from the linear-Gaussian shape were also taken into account. In order to establish PDIM data, different frame numbers between samples (Δ) were also used. Each scenario was defined for 20 scans. Also, four iterations were used for both standard H-PMHT and operational improvements applied I/H-PMHT. Thus, an optimal operation was reached, which means sufficient convergence of state estimates to target centroids was obtained without excessively increasing CPU time. Generalized scenario parameters are summarized in Table 2.

Before submitting the performance results of the tracking process for the inspected scenarios, state estimations of the algorithms and target trajectory for a particular scenario are presented. In this way we aimed to show the tracking performance after operational improvement and algorithmic amendment. Target centroids and estimation trajectories only of operational improvement applied I/H-PMHT, standard H-PMHT and

Table 2. General Summary of the Simulation Scenarios

| Scenario Number	No.of Sampling Frame (Δ)	Sensor Area (pxl²)	Area of Air Object (pxl²)	Mean Velocity (pxl/frame)	Noise Level (PDIM) (dB)	SNR (PDIM) (dB)	Noise Level (AI) (dB)	SNR (AI) (dB)
1	4	250x250	240	2.5	1	8.8	0.84	8.1
2	3	300x300	126	3.8	0.75	8.9	1.17	7.85
3	3	250x250	280	3.3	0	9.89	0.46	8.35
4	2	200x200	78	3	0	8.54	0	6.15
5	2	200x200	88	2.5	0.67	8.3	0.6	5.7
6	3	250x250	40	1.35	0.98	9.74	1.34	6.5
7	3	250x250	434	3.5	1.15	9.93	1.7	7.96
8	2	250x250	450	4.23	0	9.8	0.5	8.26
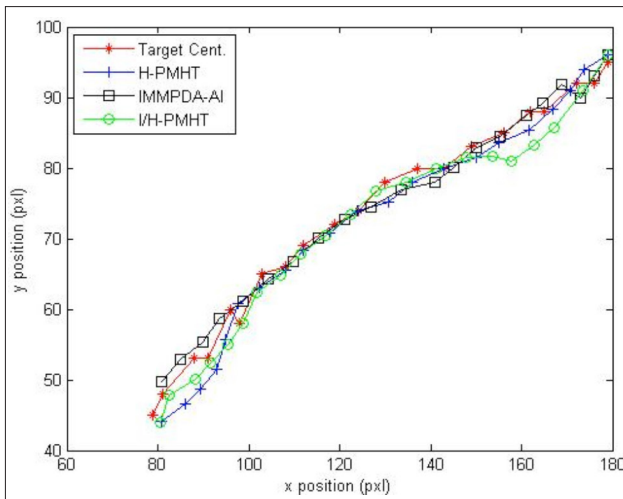
Figure 2. Target Centroids and State Estimations throughout Scenario 5 (only operational improvement is applied to I/H-PMHT)
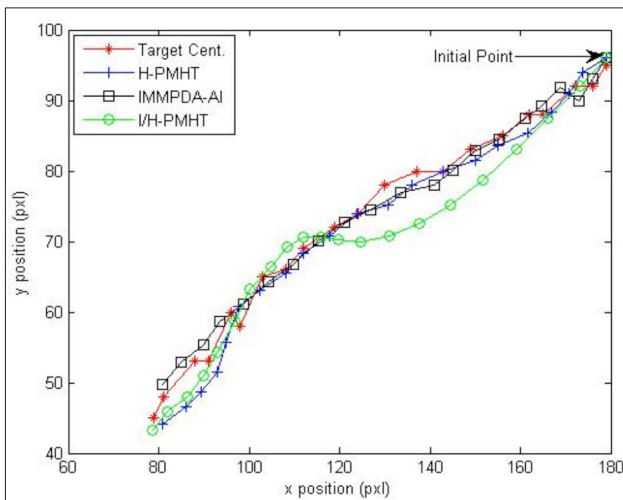

Figure 3. Target Centroids and Estimations throughout Scenario 5 (applied both operational improvements and algorithm amendment to I/H-PMHT)

IMMPDAFAI algorithms throughout the fifth scenario are given in Figure 2. The figures are given for Displayed Cell Threshold (DCT) 5 dB over noise level.

Moreover, in Figure 3 the estimation trajectories were obtained using four iterations for standard H-PMHT and one iteration for I/H-PMHT. In this case I/H-PMHT was subjected to both operational improvements and algorithm amendments, and DCT was selected 5 dB.

The performances of the algorithms are evaluated by hit on the target (HoT), and RMS position estimation error with respect to target centroid. If HoT is greater than 80%, tracking process is assumed as successful. After evaluating the performances of the algorithms, processing time is taken into account. Processing time is represented with CPU time, and this value is taken as a third

component for evaluation. The processor of the computer used in simulations is Intel-Core i5-3470 CPU with 4 cores at 3.20 GHz. The computer has 4 GB RAM, the OS is Win 7 Professional, and its instruction set is 64-bit. In Table 3 simulation results of algorithms with respect to scenarios are given for evaluation and comparison. In the table mean values of the algorithms RMS estimation errors with reference to target centroids throughout the scenarios are given as deviation from target centroid (*Dev.Trg.Cent.*).

Histogram probabilistic multi-hypothesis tracker and I/H-PMHT simulations were conducted for two different Displayed Cell Threshold (DCT) levels, which are 3 dB and 5 dB over noise level. For amplitude information of point measurement data, which is used in IMMPDA-AI simulations, AI Threshold (AIT) levels were also selected 3 dB and 5dB over noise level. A fair comparison was made between the performances of the algorithms.

The results obtained in the simulation study are summarized as follows;

- The simulation results show that standard H-PMHT and only operational improvement applied I/H-PMHT both outperform on IMMPDA-AI in terms of estimation error and HoT for whole scenarios. If we compare the results obtained with I/H-PMHT (operational improvement and algorithmic amendment applied) and the results obtained with IMMPDA-AI, they show similar performance for DCT 3 dB over noise level, and I/H-PMHT shows better performance for DCT 5 dB over noise level. In terms of processing time, IMMPDA-AI is unquestionably superior to the other algorithms. After attempting the operational improvement and algorithmic amendment, I/H-PMHT provides a respectable processing time decrease, but is still not sufficient for real-time tracking.

- When only operational improvement is applied, standard H-PMHT gives better results than I/H-PMHT at 3 dB higher level DCT, and worse results at 5 dB higher level DCT. The only exception takes place in the sixth scenario, which has a relatively small object area. These results show that only operational improvement applied I/H-PMHT (single scan, multi iteration structure) is more suitable for video object tracking than the standard H-PMHT algorithm. In this case CPU time reduction is relatively high.

- When both operational improvement and algorithmic amendment occur standard H-PMHT outperforms I/H-PMHT for 3 dB higher level DCT. On the other hand, for 5 dB higher level DCT the performance of I/H-PMHT is similar to the performance of the standard H-PMHT, and outperforms for high intensity target cases (scenarios 1, 6, 7).

- Using PDIM data with Standard H-PMHT and I/H-PMHT gives satisfactory results, and this shows that PDIM data is suitable for video object tracking.

- When the decreased processing time is taken into account from Standard H-PMHT to only operational improvement applied I/H-PMHT, and both operational improvement and algorithmic amendment applied to I/H-PMHT, then the amount of increased or decreased estimation error rate must also be tak-

Table 3. Overall Performance of Algorithms

Scenario Number	St. H-PMHT (4-Iterations)		I/H-PMHT (4-Iterations) Only Opr.Improvement		I/H-PMHT (1-Iteration) Both Opr.Imp.and Algo. Amendment		IMMPDA-AI	
	DCT 3dB	DCT 5dB	DCT 3dB	DCT 5dB	DCT 3dB	DCT 5dB	AIT 3dB	AIT 5dB
1	D=4 H=100 C=539.7	D=4.04 H=100 C=547	D=5.3 H=85 C=234.9	D=2.6 H=100 C=238	D=10.6 H=70 C=185.7	D=3.95 H=100 C=191	D=10.2 H=80 C=0.65	D=10.7 H=80 C=0.23
2	D=3.5 H=100 C=1190.7	D=2.78 H=100 C=1163	D=3.75 H=90 C=777	D=1.6 H=100 C=726.3	D=6 H=80 C=673	D=3.13 H=100 C=670.6	D=11.8 H=15 C=0.8	D=11.8 H=30 C=0.8
3	D=7.2 H=85 C=533.5	D=4.1 H=95 C=533	D=8.4 H=80 C=236	D=4.27 H=90 C=235	D=11.4 H=70 C=194.2	D=4.34 H=90 C=185.6	D=10 H=100 C=0.5	D=10 H=100 C=0.5
4	D=7 H=70 C=303.3	D=1.8 H=100 C=301.2	D=7.4 H=65 C=111.2	D=1.6 H=100 C=112.1	D=10.2 H=50 C=80.2	D=4.5 H=100 C=79.7	D=7.8 H=65 C=0.51	D=8.25 H=60 C=0.51
5	D=5.3 H=80 C=294.6	D=2.14 H=100 C=297.5	D=5.7 H=80 C=110	D=2.9 H=100 C=110.8	D=9.5 H=60 C=78.6	D=7.26 H=70 C=78.9	D=6 H=85 C=0.5	D=6 H=80 C=0.36
6	D=4.8 H=75 C=530.4	D=5.1 H=75 C=534	D=4.2 H=85 C=235	D=2.2 H=100 C=238.4	D=6 H=75 C=186.7	D=3.6 H=90 C=186.1	D=4.37 H=95 C=0.5	D=4.4 H=100 C=0.5
7	D=4.5 H=100 C=529.8	D=4.34 H=100 C=528.4	D=8.6 H=85 C=236.3	D=3.37 H=100 C=234.7	D=9.4 H=85 C=185.9	D=3.65 H=100 C=185.6	D=9.5 H=100 C=0.46	D=9.47 H=95 C=0.47
8	D=4.23 H=100 C=532.7	D=3.6 H=100 C=535.6	D=4.4 H=100 C=236.4	D=3.7 H=100 C=235.5	D=4.5 H=100 C=186.7	D=4.1 H=100 C=185.5	D=6.5 H=100 C=0.53	D=6.5 H=100 C=0.54

D: mean deviation from target centroid throughout tracking process (in pixels); H: hit on target-HoT throughout tracking process (%); C: CPU time usage throughout tracking process (in seconds); DCT 3 dB: displayed cell threshold is in the level of 3 db higher than noise level; DCT 5 dB: displayed cell threshold is in the level of 5 db higher than noise level; AIT: amplitude information threshold

en into account. For this purpose, Table 4 was prepared using the results of the simulation study.

Table 4 shows that a decrease in processing time is directly related to the sensor area. In wider sensor areas processing time decreasing is low, and vice versa. Also converging to target centroids with standard H-PMHT occurs rapidly for small sensor areas (scenarios 4 and 5), even if target intensities are not high. I/H-PMHT requires high target intensities, and especially for DCT's 5 dB higher than noise level has sufficient success. Estimation errors of I/H-PMHT obtained for all scenarios are within acceptable limits for track continuation.

Conclusion

This study has provided a new approach for reducing the CPU time of H-PMHT algorithm. For this purpose, operational im-

provement and algorithmic amendment processes were applied to standard H-PMHT algorithm, and I/H-PMHT algorithm was obtained. Measurement data used in the study is defined as PDIM data. The PDIM data was obtained from true color video data of flying objects taken from the ground. PDIM data is very useful and easily processed for filtering moving objects from the stationary ones, and for use with H-PMHT and its variations.

In the experimental study using PDIM data, standard H-PMHT and operational improvement applied I/H-PMHT and both operational improvement and algorithmic amendment applied I/H-PMHT algorithms present satisfactory tracking results. The comparison was made between standard H-PMHT, operational improvement applied I/H-PMHT, and operational improvement and algorithmic amendment applied I/H-PMHT algorithms. Furthermore IMMPDA-AI algorithm was used for comparison purposes. The results show that improvement on processing

Table 4. Offset Compromise between Processing Time and Performance of Standard and Improved H-PMHT Algorithms

Scn. No.	Sensor Area pxl2	CPU Time Decrease After Opr. Imp. (%)		CPU Time Decrease After Algo.Amdt. (%)		Total CPU Time Decrease (%)		Total Estimation Error Rate (D_{St}/D_{Imp})	
		DCT 3dB	DCT 5dB	DCT 3dB	DCT 5dB	DCT 3dB	DCT 5dB	DCT 3dB	DCT 5dB
1	250x250	56.5	56.5	20.9	19.7	65.6	65.1	0.38	1.02
2	300x300	34.7	37.5	15.4	7.7	43.4	42.3	0.58	0.88
3	250x250	55.7	56	17.7	21	63.6	65.2	0.63	0.94
4	200x200	63.3	62.8	28	29	73.5	73.5	0.68	0.4
5	200x200	62.6	62.7	28.5	28.8	73.3	73.4	0.56	0.29
6	250x250	55.7	55.3	20.5	22	64.8	65.1	0.8	1.41
7	250x250	55.4	55.6	21.3	21	64.9	64.8	0.48	1.19
8	250x250	55.6	56	21	21.2	65	65.4	0.94	0.87

D_{St}: mean deviation of standard H-PMHT (in pixels); D_{Imp}: mean deviation of I/H-PMHT-both operational improvement and algorithmic amendment applied (in pixels); Opr. Imp.: operational improvement; Algo. Amdt.: algorithmic amendment

time up to 73.5% (with respect to sensor area dimensions) is possible. Also, further reducing the sensor area will result in further decrease in processing time. The tracking performance after improvements is also sufficient for tracking continuity.

In this work processing time reduction was achieved, but these results are still not sufficient for real time video tracking. It is assumed that without changing the structure of H-PMHT the lowest CPU time usage border is reached. In the light of these indications future work is planned in which an FPGA based technique in addition to I/H-PMHT will be used in order to achieve real time tracking. In this study only single-target scenarios have been taken into account, however in some practical circumstances multiple target tracking is required. Also implementing multi-target tracking with I/H-PMHT using PDIM data is another subject for future work.

Peer-review: Externally peer-reviewed.

Conflict of Interest: The author have no conflicts of interest to declare.

Financial Disclosure: The author declared that the study has received no financial support

References

1. S. J. Davey, M. G. Rutten, B. Cheung, "A Comparison of Detection Performance for Several Track-before-Detect Algorithms", *EURASIP J. Adv. Signal Process*, 2008.
2. C. Kathirvel, D. Mohanapriya, K. Mahesh, "A Novel Robust Approach for Moving Object Detection and Tracking in Video Surveillance System", *IJSRST*, vol. 3, no. 8, pp.1235-1241, 2017.
3. X, Dong, J. Shen, D. Yu, W. Wang, J. Liu, H. Huang, "Occlusion-Aware Real-Time Object Tracking", *IEEE Transactions On Multimedia*, vol. 19, no. pp. 763-771, 2017
4. S. S. Blackman, R. Popoli, "Design and Analysis of Modern Tracking Systems", Artec House, Norwood, MA, USA, 1999.
5. Y. Bar-Shalom, X. R. Li, "Multitarget-Multisensor Tracking: Principles and Techniques", YBS, Storrs, CT, USA, 1995.
6. R. L. Streit, "Tracking on Intensity-Modulated Data Streams, NUWC-NPT Technical Report 11.221, Naval Undersea Warfare Center Division" Newport, Rhode Island, USA, 2000.
7. H. X. Vu, S. J. Davey, "Track-Before-Detect using Histogram PMHT and Dynamic Programming, In Digital Image Computing Techniques and Applications", International Conference (DICTA 2012), Fremantle, WA, 2012.
8. M. J. Walsh, M. L. Graham, R.L. Streit, T.E. Luginbuhl, L. E. Mathews, "Tracking on Intensity-Modulated Data Streams, Proceedings of the 2001", IEEE Aerospace Conference, Big Sky, MT, 2001.
9. R. L. Streit, M.L. Graham, M. J. Walsh, "Tracking in Hyper-Spectral Data, Information Fusion 2002", Annapolis, MD, USA, 2002, pp.852-859.
10. A. G. Pakfiliz, M. Efe, "Multi-Target Tracking in Clutter with Histogram Probabilistic Multi-Hypothesis Tracker, In Proc." IEEE Conf Syst Eng, Las Vegas, USA, pp.137-142, 2005.
11. S. J. Davey, "Histogram PMHT with Particles", In Proceedings of the 14th International Conference on Information Fusion", Chicago, Illinois, USA, 2011.
12. M. Wieneke, S. J. Davey, "Histogram PMHT with Target Extent Estimates Based on Random Matrices", In Proceedings of the 14th International Conference on Information Fusion, Chicago, Illinois, USA, July 2011.
13. Y. Bar-Shalom, T. Kirubarajan, X. Lin, "Probabilistic Data Association Techniques for Target Tracking with Applications to Sonar, Radar and EO Sensors", *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 8, pp.37-56, 2005.
14. R. C. Gonzalez, R.E. Woods, "Digital Image Processing. 2nd ed. Prentice Hall, New Jersey (USA), 2002.
15. A.J. Lipton, H. Fujiyoshi, R. S. Patil, "Moving target Classification and Tracking from Real-Time Video". Proc of Workshop Applications of Computer Vision, Princeton, New Jersey, USA, pp. 129-136, 1998.
16. C. Stauffer, W. Eric, L. Grimson, "Learning Patterns of Activity Using Real-Time Tracking". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 747-757, 2000.

17. Y. Boers, F. Ehlersand, W. Koch, T. Luginbuhland, L. D. Stone, R. L. Streit, "Track before Detect Algorithms", *EURASIP J Adv Signal Process*, 2008.
18. S. J. Davey, H. X. Gaetjens, "Track-Before-Detect Using Expectation Maximisation: The Histogram Probabilistic Multi-hypothesis Tracker: Theory and Applications", Springer, Singapore, 2018.
19. S. J. Davey, M. Wieneke, "Tracking Groups of People in Video with Histogram-PMHT" Defense Apps. of Signal Processing (DSAP) 2011 Workshop, Coolum, Queensland, Australia, 2011.
20. R. L. Streit, T. E. Luginbuhl, "Probabilistic Multi Hypothesis tracking, Technical Report NUWC-NPT 10,428, Naval Undersea Warfare Center Division", Newport, Rhode Island, USA, 1995.
21. Y. Bar-Shalom, T.E. Fortmann, "Tracking and Data Association" Academic Press, NY, USA, 1988.

Ahmet Güngör Pakfiliz received his B.S. degree in Electrical Engineering from Yıldız Technical University in 1991, M.S. degree in Electrical and Electronics Engineering from Middle East Technical University in 1997 and Ph.D. degree in Electronics Engineering from Ankara University in 2004. He is currently an Assistant Professor at the Department of Electrical and Electronics Engineering of Başkent University.