# Data Association With Camera Parameters Estimation for Object Tracking From Drones

**ZIJIAO TIAN**
**YAAKOV BAR-SHALOM**
**RONG YANG**
**HONG'AN JACK HUANG**
**GEE WAH NG**

**This paper considers the problem of inaccurate measurement-to-track association (M2TA) and poor tracking caused by camera motion changes in drone-captured video. The camera often changes its field of view to track targets; however, the sudden change leads to inaccurate M2TA and degrades tracking performance. Previous work estimated the 3D camera motion parameter vector (zoom ratio, panning, and tilting) and associated measurements and tracks only between two consecutive frames. This paper extends the camera motion parameter to 4D by including rolling and sequentially associates (forward) measurements to tracks over the entire data. The estimated camera parameters improve the predicted measurements and achieve better M2TA. Results on real data illustrate the benefits of the proposed method (association with 4D camera parameters estimation) that yields better associations and improves tracking accuracy compared to the state-of-the-art gating method based on inflated covariances.**

Manuscript received December 26, 2023; revised August 21, 2024; released for publication October 14, 2024

Z. Tian and Y. Bar-Shalom are with the Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269, USA (e-mail: zijiao.tian@uconn.edu; yaakov.bar-shalom@uconn.edu).

R. Yang and H. A. J. Huang are with the DSO National Laboratories, Singapore 118225 (e-mail: yrong@dso.org.sg; hhongan@dso.org.sg).

G. W. Ng is a staff from DSO National Laboratories Singapore 118225, who is on secondment to Home Team Science and Technology Agency, Singapore 138507 (e-mail: ng_gee_wah@htx.gov.sg).

1557-6418/2024/$17.00 

## I. INTRODUCTION

Unmanned aerial systems (UAS), equipped with cameras, are extensively used to capture images and videos for tracking systems. These systems are crucial for surveillance applications. For example, UAS can be employed to monitor borders to detect illegal crossings or smuggling activities, as well as to observe traffic flow to identify and respond to accidents. These cameras often adjust their field of view (FoV) to keep up with moving targets [12], [16]–[18], [24]. However, changes in pointing and/or in image scale make the target data association and tracking from drones more challenging than traditional data association and object tracking. The camera movements, such as pan (yaw), tilt (pitch), zoom, and platform roll, are not available to the data association and tracking algorithms and can degrade the reliability of the video tracks [7], [15], [19].

Camera vibrations or movements will introduce instability into video images, posing challenges in video image stabilization and registration. Most methods address this problem by compensating camera motion with camera motion estimation. Typically, camera motion estimation is classified into two categories: intensity-based motion estimation and feature-based motion estimation. The intensity-based motion estimation (such as using image grayscale [14], phase correlation [10], pixel-based correlation [11]) is based on pixel intensities between consecutive frames, while the feature-based motion estimation (such as edges and corners [6]) is based on extracting and matching features across consecutive frames. Some advanced video stabilization techniques combine both approaches in [28].

In [23], it was shown that the tilt, pan, and roll angle errors of the camera can affect the navigational parameters in autonomous vehicles. Allebosch et al. [1] compensated camera motion by estimating panning and tilting with different models. A reversible jump Markov chain Monte Carlo method was employed for estimating camera parameters consisting of angles and positions in [8]. Our previous work [25] assumes that the camera motion parameters are described by zoom ratio, panning, and tilting of the focal-plane array. The 3D camera parameter vector is directly solved by the MLE3 (maximum likelihood estimation in 3D) approach via linear least squares (LLS). However, this does not account for UAS camera rolling. Moreover, it was tested only between two consecutive frames and was not applied to the entire duration of the data [25]. Thus, it could not provide an evaluation of the tracking accuracy during an entire video sequence.

In this work, the goal is to develop an approach for accurate tracking in the presence of camera panning, tilting, zooming, and rolling for drone-captured video. Targets are detected by a state-of-the-art object detection algorithm—You Only Look Once (YOLO) [20], which provides bounding boxes (BBs). To track accurately, the camera has the capability to adjust its view,

and the UAS can change its velocity and altitude. This is accomplished visually by a human operator. The attitude is operator-controlled but not quantified to be usable for the algorithm. The camera motion parameters, unknown to the operator or the tracker, are represented by a 4D vector including zoom ratio, panning, tilting, and rolling. These parameters are estimated using the MLE4 (maximum likelihood estimation in 4D) via the iterated least squares (ILS) method in each frame. Based on the estimated camera parameters, one associates the measurement-track pairs, and then corrected state predictions are calculated. These corrected state predictions play an important role in the filtering step (Kalman filter), resulting in more accurate state updates. Compared with the conventional measurement-to-track association (M2TA), the validation gating method, which relies on inflated measurement covariance for association, the proposed algorithm has superior robustness and performance. If there are unexpected target state changes due to camera movements, the measurements may not be correctly associated with their tracks by the gating method. In contrast, our method has enhanced robustness by integrating camera motion parameter estimation into the association and tracking process. This integration allows our algorithm to maintain accurate tracking even in the presence of abrupt camera motion changes, which would typically challenge the conventional gating method.

The contributions of this paper are as follows: Firstly, it extends the camera motion parameter vector from 3D to 4D, enhancing its capability not only for zoom, pan, and tilt, but also for roll movements. The 3D parameters are estimated using MLE3, while the 4D parameters are estimated using MLE4. The latter is shown to have smaller errors in tracking results compared to 3D parameter vector estimation. Additionally, the proposed method can handle scenarios with both a large and small number of targets with different assignment methods. It adapts effectively to diverse tracking environments.

The rest of the paper is structured as follows: Section II introduces the target detection by YOLO, presents the baseline Gating Method with Inflated Covariance (GMIC) method, and outlines the overall system flow. Section III presents the estimation of the camera motion parameters consisting of both the 3D vector and the 4D vector. Section IV presents the proposed approach that integrates camera parameters into the association process. Section V shows the real data results and discusses them. Section VI draws the conclusions.

## II. PROBLEM FORMULATION

This section first introduces the target position detection from YOLO in each frame. Then the baseline M2TA method is briefly presented, which is used to compare with the proposed method. The tracking system flow chart is later shown in this section. After target detection by YOLO, the association is combined with camera parameter estimates to yield better target state predictions.

### A. YOLO: Target Detection

A review of deep learning applied to computer vision for target detection can be found in [29]. One of the most popular algorithms is YOLO [20]. YOLO integrates feature extraction, object localization, BB regression, and classification in a monolithic network. It maps from image pixels to BB coordinates and class probabilities. The basic idea is dividing the input image into a grid, and each grid cell is responsible for predicting the object. It uses a single-stage architecture to make predictions for multiple objects, making it faster and more efficient than traditional object detection algorithms. YOLO v3 [21], applied on a per frame basis, is an incremental improvement over YOLO in detection and BB accuracy for smaller targets. Besides, its efficiency in processing time makes it widely used in real-time object detection.

In this paper, the targets are detected by YOLO v3 in each frame and the measurements are the target's positions, specifically, the top left corner of the BB. Although it can identify multiple objects from a video frame and label them with corresponding class probabilities, we focus on people and their 2D position information.

### B. Gating Method With Inflated Covariance (GMIC)

The traditional state-of-the-art M2TA method—validation gate method—assumes that the target motion can be utilized to predict the "measurement association regions" [3], [4], [27]. It eliminates unlikely measurements that need to be considered for association with a track [22], [26]. The gating method is based on inflated covariances. The camera parameter changes create bias in the estimation. Instead of estimating the bias, this approach increases the measurement noise standard deviations to "cover" the bias implicitly. For the association, the association gate is enlarged by an inflated measurement error variance so that shifted measurements can be associated with their tracks.

Consider a set of measurements $z_i(k)$ at time (frame index) $k$, $i = 1, 2, \ldots, N_m(k)$, where $N_m(k)$ is the number of measurements at time $k$, and a set of predicted measurements $\hat{z}_j(k|k-1)$ at time $k$, $j = 1, 2, \ldots, N_t(k)$, where $N_t(k)$ is the number of tracks for which a prediction at time $k$ is available. The difference between each actual and predicted measurement (the filter innovation) is defined as

$$\tilde{n}_{ij}(k) = z_i(k) - \hat{z}_j(k|k-1). \tag{1}$$

A gate is formed about the predicted measurement, and all actual measurements (observations) that fall within the gate are considered for track updates. Define a gate threshold $\gamma$ such that association is allowed if the norm

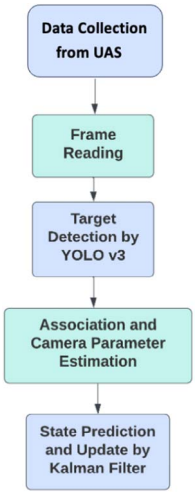



Figure 1. Flow chart of the tracking system.

of the residual falls within the gate

$$d_{ij}^2(k) = \tilde{n}'_{ij}(k) S_j(k)^{-1} \tilde{n}_{ij}(k) \leq \gamma, \tag{2}$$

where $d_{ji}^2$ is also known as the squared Mahalanobis distance between track $j$ and measurement $i$ and $S_j(k)$ is the corresponding innovation covariance. Once the potential measurements are chosen based on (2), they are associated to tracks using the Auction method. In this process, each track "bids" for the measurements, and the goal is to find the highest "bid" track and then assign the measurement to the track. The details can be found in [4].

In target tracking, the assigned measurements are incorporated into the updated track state estimates during the filtering step. However, it is important to note that gating, while a commonly used heuristic method, is not infallible. Due to sudden camera movements such as panning, zooming, tilting, or rolling, the validation gates may lead to incorrect M2TA and poor tracking performance, even in the presence of inflated gates.

### C. System Flow for Assignment With Camera Parameter Estimation

The overall system is illustrated in the flow chart in Fig. 1. It outlines several key steps, including data collection, target detection, a novel approach to association that incorporates camera parameter estimation, and filtering. Compared with the previous gating method, the main difference is that we integrate camera parameter estimation into the association process. This mitigates the effects of sudden camera movements, thus yielding more accurate M2TA and better state prediction. The proposed algorithm sequentially estimates the camera state and the target states at each time step, which is similar to the simultaneous localization and mapping (SLAM) [5]. However, SLAM typically focuses on building a map in a static environment, while our problem focuses on tracking targets whose locations in the FoV change due to camera parameter variation. Besides, our approach has to be implementable in real time with modest computing requirements; adding moving targets as in SLAM would significantly increase these requirements.

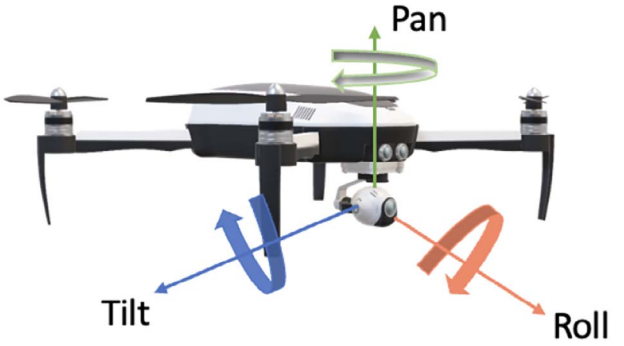



Figure 2. Camera has pan, tilt, and roll movements.

As shown in Fig. 2, the camera mounted on drone gimbals can be controlled by servo driver modules. When it comes to recording of image frames, the less vibration and camera shake the better. However, sudden camera movements are inevitable in practical scenarios. Specifically, we investigate the several camera motion parameters:

1. Pan (Yaw): A yaw motion describes the left and right (horizontal) movement of the camera.

2. Tilt (Pitch): An up and down (vertical) movement of the camera.

3. Roll: A roll motion is a rotation around the camera axis direction. The roll comes from "banking" when the UAS turns.

It should be pointed out that the commands for the above are given usually by a human operator, but they are not measured or available to the data association or tracking algorithms.

Additionally, the camera's field of view can be adjusted by zooming in or out. Although zoom is not a camera position movement, one can change its focal length to change image size. These camera motion parameters are crucial for the quality of tracking and will be estimated in the subsequent section.

## III. CAMERA MOTION PARAMETERS ESTIMATION

This section presents the formulation and estimation of the camera parameters using 3D and 4D vectors when the camera has sudden motion change. The corrected target position prediction conditioned on the parameter vector estimates is also given.

### A. 3D Camera Parameters Estimation (MLE3)

The 3D camera parameter vector consisting of the zoom ratio $\phi(k)$, camera panning (horizontal motion) $x_c(k)$, and tilting (vertical motion) $y_c(k)$ is given by

$$\zeta(k) = \begin{bmatrix} \phi(k) \\ x_c(k) \\ y_c(k) \end{bmatrix}. \tag{3}$$

The camera parameter vector (3) [or (5) below] is assumed an unknown constant for each $k$, which is the model for the least squares [2] (LS) algorithm. No dynamics are assumed across time since the goal was simplicity. Using a dynamic model for the camera motion (and a Kalman filter to estimate it) is possible, but it would increase the computation complexity, and since the LS worked very well, there was no need to increase the complexity.

Assume there is a set of measurements $\mathbf{z}_i(k)$ at time $k, i = 1, 2, \ldots, N_m$, and a set of predicted measurements $\hat{\mathbf{z}}_j(k|k-1)$ at time $k$, $j = 1, 2, \ldots, N_t$. The corrected prediction (denoted by hat and superscript "$\kappa$") conditioned on the 3D parameter vector is[1]

$$\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \boldsymbol{\zeta}(k)] = \begin{bmatrix} \hat{x}_j(k|k-1)\phi(k) + x_c(k) \\ \hat{y}_j(k|k-1)\phi(k) + y_c(k) \end{bmatrix}, \quad (4)$$

where $\hat{x}$ and $\hat{y}$ are the predicted position of the target. The 3D camera parameter vector without rolling is solved by MLE (designated as MLE3) via LLS. The details are shown, for completeness, in Appendix A (based on [25]).

## B. 4D Camera Parameter Estimation (MLE4)

The 4D camera parameter vector including the rolling $\rho(k)$ is denoted as

$$\boldsymbol{\xi}(k) = \begin{bmatrix} \rho(k) \\ \phi(k) \\ x_c(k) \\ y_c(k) \end{bmatrix}, \quad (5)$$

where $\phi$ is the zoom ratio, which can be expressed as the ratio of focal length $f(k)$ at different time as follows[2]:

$$\phi(k) = \frac{f(k)}{f(k-1)}. \quad (6)$$

Then, the corrected prediction conditioned on the 4D camera parameter vector is given by

$$\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \boldsymbol{\xi}(k)] = \begin{bmatrix} [\hat{x}_j(k|k-1)\cos\rho(k) + \hat{y}_j(k|k-1)\sin\rho(k)]\phi(k) + x_c(k) \\ [\hat{y}_j(k|k-1)\cos\rho(k) - \hat{x}_j(k|k-1)\sin\rho(k)]\phi(k) + y_c(k) \end{bmatrix}. \quad (7)$$

Define $i(j)$ as the index of the measurement associated with track $j$. The measured target position is denoted by

$$\mathbf{z}_{i(j)}(k) = \begin{bmatrix} h_1[\hat{x}_{i(j)}(k|k-1), \hat{y}_{i(j)}(k|k-1), \boldsymbol{\xi}(k)] + n_{i(j),1}(k) \\ h_2[\hat{y}_{i(j)}(k|k-1), \hat{x}_{i(j)}(k|k-1), \boldsymbol{\xi}(k)] + n_{i(j),2}(k) \end{bmatrix}, \quad (8)$$

with

$$h_1 = [\hat{x}_{i(j)}(k|k-1)\cos\rho(k) + \hat{y}_{i(j)}(k|k-1)\sin\rho(k)]\phi(k) + x_c(k), \quad (9)$$

$$h_2 = [\hat{y}_{i(j)}(k|k-1)\cos\rho(k) - \hat{x}_{i(j)}(k|k-1)\sin\rho(k)]\phi(k) + y_c(k), \quad (10)$$

where $n_{i(j),\ell}(k), \ell = 1, 2$ are mutually independent zero-mean white Gaussian residuals with variance $\sigma^2$. The 4D camera parameters are estimated by the MLE4 algorithm via ILS estimator [2]. The ILS recursion is as follows: Using a first order series expansion about $\hat{\boldsymbol{\xi}}$, one has

$$\mathbf{z}_{i(j)}(k) = \begin{bmatrix} h_1[\hat{x}_{i(j)}(k|k-1), \hat{y}_{i(j)}(k|k-1), \hat{\boldsymbol{\xi}}(k)] \\ h_2[\hat{y}_{i(j)}(k|k-1), \hat{x}_{i(j)}(k|k-1), \hat{\boldsymbol{\xi}}(k))] \end{bmatrix} \quad (11)$$

$$+ \begin{bmatrix} \boldsymbol{J}_1(\hat{\boldsymbol{\xi}}(k) - \hat{\boldsymbol{\xi}}(k+1)) + n_{i(j),1}(k) \\ \boldsymbol{J}_2(\hat{\boldsymbol{\xi}}(k) - \hat{\boldsymbol{\xi}}(k+1)) + n_{i(j),2}(k) \end{bmatrix}. \quad (12)$$

We can define the following matrices:

$$\mathbf{h}_{i(j)} = [h_1 \;\; h_2]' \quad (2 \times 1), \quad (13)$$

$$\mathbf{h} = \begin{bmatrix} \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_N \end{bmatrix} \quad (2N \times 1), \quad (14)$$

$$\Re_{i(j)} = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix} \quad (2 \times 2), \quad (15)$$

$$\mathbf{R} = \text{diag}[\Re_1 \ldots \Re_N] \quad (2N \times 2N), \quad (16)$$

$$\mathbf{z}_{i(j)} = [x_{i(j)} \;\; y_{i(j)}]' \quad (2 \times 1), \quad (17)$$

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_N \end{bmatrix} \quad (2N \times 1), \quad (18)$$

where $N$ is the number of M2TA pairings, and $i(j) = 1, \ldots, N$. The Jacobian matrix $\mathbf{J}$ is given by

$$\mathbf{J} = [\mathbf{H}_1 \;\; \mathbf{H}_2 \;\; \ldots \;\; \mathbf{H}_N]' \quad (2N \times 4), \quad (19)$$

where

$$\mathbf{H}_j = \begin{bmatrix} \frac{\partial h_1}{\partial \rho} & \frac{\partial h_1}{\partial \phi} & \frac{\partial h_1}{\partial x_c} & \frac{\partial h_1}{\partial y_c} \\ \frac{\partial h_2}{\partial \rho} & \frac{\partial h_2}{\partial \phi} & \frac{\partial h_2}{\partial x_c} & \frac{\partial h_2}{\partial y_c} \end{bmatrix} \quad (2 \times 4). \quad (20)$$

The partial derivatives are shown in Appendix B.

Finally, the updated ILS estimates $\hat{\boldsymbol{\xi}}(k+1)$ $(4 \times 1)$ is then obtained as

$$\hat{\boldsymbol{\xi}}(k+1) = \hat{\boldsymbol{\xi}}(k) + (\boldsymbol{J}'(k)\boldsymbol{R}^{-1}\boldsymbol{J}(k))^{-1}\boldsymbol{J}'(k)\boldsymbol{R}^{-1}[\boldsymbol{z}(k) - \boldsymbol{h}(\hat{\boldsymbol{\xi}}(k))]. \quad (21)$$

[1] Note that (4) is written with the yet to be estimated camera parameters.

[2] The focal lengths are unknown and not observable (unless the sizes of the targets are known and the noises are much smaller). However, the ratio (6) can be estimated together with the association.

## IV. ASSOCIATION WITH CAMERA PARAMETER ESTIMATION

This section first introduces the target motion model. Next, the association with estimation methods in terms of the number of M2TA pairs is presented. The filtering step using the Kalman filter is also provided.

### A. Dynamic Models

There are multiple targets in the observed frame, and they are assumed to move with a nearly constant velocity (NCV). The target motion model is characterized by a continuous white-noise acceleration (CWNA) model [2]. The state vector consisting of position and velocity in the camera image is

$$\mathbf{x}(k) = [x(k) \;\; y(k) \;\; \dot{x}(k) \;\; \dot{y}(k)]'. \tag{22}$$

For sampling interval $T$, the state and measurement equations are

$$\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{v}(k), \tag{23}$$

$$\mathbf{z}(k) = \mathbf{H}\mathbf{x}(k) + \mathbf{w}(k), \tag{24}$$

where

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{25}$$

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{26}$$

The measurements consist of the state's position components. For the above, $\mathbf{v}(k)$ is the zero mean white process noise sequence with covariance

$$\mathbf{Q}(k) = \begin{bmatrix} \frac{T^3}{3} & 0 & \frac{T^2}{2} & 0 \\ 0 & \frac{T^3}{3} & 0 & \frac{T^2}{2} \\ \frac{T^2}{2} & 0 & T & 0 \\ 0 & \frac{T^2}{2} & 0 & T \end{bmatrix} q, \tag{27}$$

where $q$ is the process noise power spectral density (assumed the same in $x$ and $y$) and $\mathbf{w}(k)$ is the zero mean white measurement noise sequence with covariance $\mathbf{R}(k) = \text{diag}[\sigma^2 \; \sigma^2]$.

### B. Association With Estimation

The association should be done between the following:

1. Tracks represented by corrected predictions $\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \boldsymbol{\zeta}(k)]$, $j = 0, 1, \ldots, N_t$ (with $\boldsymbol{\zeta}$ or $\boldsymbol{\xi}$ to be estimated),[3] where the index $j = 0$ represents the "dummy tracks" to which the unassociated measurements belong.

2. Measurements $\mathbf{z}_i(k)$, $i = 0, 1, \ldots, N_m$, where the index $i = 0$ represents the "dummy measurements" to which the unassociated tracks belong.

The cost of assigning $\mathbf{z}_i(k)$ to $\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \boldsymbol{\zeta}(k)]$ is the negative log-likelihood function [2] (scalar normalized squared distance)[4]

$$c[i, j, k, \boldsymbol{\zeta}(k)] = ||\mathbf{z}_i(k) - \hat{\mathbf{z}}_j^{\kappa}(k|k-1)||^2. \tag{28}$$

Initial candidate measurement-to-track pairs are based on the GMIC method. There are two different association methods in terms of the number of measurement-to-track pairs:

Method A: When the number of pairs is small, the first iteration estimates camera parameters is based on the first set of measurement-to-track pairs. Then it corrects the predictions. The second iteration of the assignment is based on other pairs of measurements and tracks. The iteration is stopped until all combinations are exhausted. Finally, we choose the assignment that yields the lowest cost along with its estimated parameters.

This method can find the optimal measurement-to-track pairs, but requires a global minimization (exhaustive) search of all the combinations of measurements and tracks. It requires $O(n!)$ operations, where $n$ is the number of the measurement-to-track pairs. Due to this complexity, it is practical only for a limited number of pairs. Typically this is feasible for $n < 6$.

Method B: To enhance computational efficiency for $n \geq 6$, the estimation is combined with the 2D assignment algorithm, specifically the Auction or Hungarian method[5] [9]. The procedure is as follows: The 2D assignment algorithm first finds the assignment of measurements to tracks that minimizes the total cost. Then the camera parameter vector is estimated based on this assignment, the predictions are corrected, and the target state is updated. This method requires $O(n^3)$ running time, making it efficient for a large number of tracks and measurements.

To combine the above methods, when the number of pairs is small ($n! \leq n^3$, for $n < 6$), the first estimation method with global search is utilized. Otherwise, when dealing with a large number of pairs, the estimation with the 2D assignment algorithm is preferred to ensure computational efficiency.

[3]The camera parameter estimation for $\boldsymbol{\zeta}$(3D) or $\boldsymbol{\xi}$(4D) is the same except for their dimensions, so we use the notation $\boldsymbol{\zeta}$ in Section IV.

[4]We assume the innovation covariances are all diagonal and equal, thus we can omit them.

[5]This article focuses on the camera estimation rather than the assignment algorithms. The chosen assignment algorithms are simple for real-time implementation, but one can also use other 2D assignment algorithms.

Table I
The Algorithms Considered in the Paper

| Acronym | Algorithm |
|---|---|
| GMIC | Gating method with inflated covariances |
| MLE3 [25] | ML estimator for 3D camera parameter vector via LLS |
| MLE4 | ML estimator for 4D camera parameter vector via ILS |

### C. Predictions

After association and camera parameter estimation, the assigned measurements are corrected and used to obtain the updated track state estimates during the filtering stage. The improved (position) prediction $\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \hat{\boldsymbol{\zeta}}(k)]$ is used for the target state update in the Kalman filter. Thus, the updated state estimate $\hat{\mathbf{x}}$ for target $j$ and the updated covariance are given by[6]

$$\hat{\mathbf{x}}_j(k|k) = \hat{\mathbf{x}}_j^{\kappa}[k|k-1, \hat{\boldsymbol{\zeta}}(k)] + \mathbf{K}(k)\boldsymbol{\nu}_j(k), \tag{29}$$

$$\mathbf{P}_j(k|k) = \mathbf{P}_j(k|k-1) - \mathbf{K}(k)\mathbf{S}(k)\mathbf{K}'(k), \tag{30}$$

where the filter gain $\mathbf{K}(k)$ and innovation covariance are

$$\mathbf{K}(k) = \mathbf{P}_j(k|k-1)\mathbf{H}'(k)\mathbf{S}^{-1}(k), \tag{31}$$

$$\mathbf{S}(k) = \mathbf{H}(k)\mathbf{P}_j(k|k-1)\mathbf{H}'(k) + \mathbf{R}(k), \tag{32}$$

the measurement residual (innovation) is

$$\boldsymbol{\nu}_j(k) = \mathbf{z}_j(k) - \hat{\mathbf{z}}_j^{\kappa}[k|k-1, \hat{\boldsymbol{\zeta}}(k)]. \tag{33}$$

Since the proposed method (2D assignment with camera parameter estimation) will be shown to provide a more accurate prediction, the tracking results will be better than the validation GMIC method when the camera is panning, tilting, zooming or rolling.

## V. REAL DATA RESULTS

Two types of real scenarios are considered: with a small number of targets (using exhaustive search) and with a large number of targets (using Hungarian assignment algorithm). MLE4 is compared with MLE3 and GMIC, see Table I.

### A. Initialization

The initial candidate associations for both MLE3 and MLE4 are based on GMIC. GMIC merges the nearby YOLO measurements when there is a one-pixel difference. If the merging distance is large, GMIC will make wrong associations, especially for tracks lacking measurements. To get a better association, we only focus on the confirmed tracks, and remove the inactive tracks (for example, previous tracks that moved out of FoV).

[6]The prediction $\hat{\mathbf{x}}_j^{\kappa}[k|k-1, \hat{\boldsymbol{\zeta}}(k)]$ has position components $\hat{\mathbf{z}}_j^{\kappa}[k|k-1, \hat{\boldsymbol{\zeta}}(k)]$, and its velocity components are unchanged from $\hat{\mathbf{x}}_j(k|k-1)$.

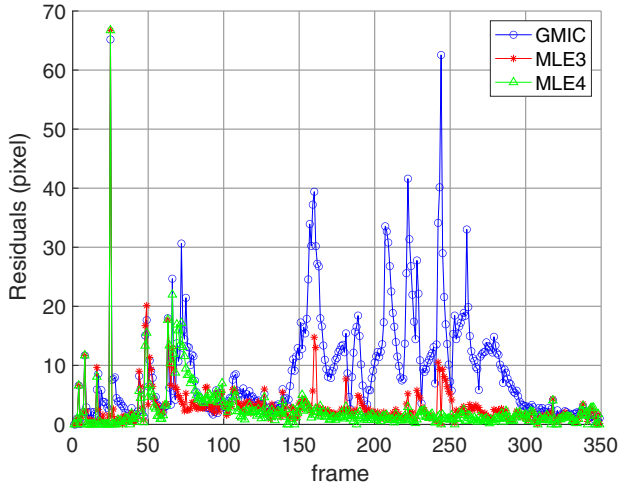



Figure 3. HTX scenario: Average track residual errors for GMIC, MLE3, and MLE4 are 9.04, 2.94, and 2.48, respectively.

The new tracks are initialized such that the first estimate $\hat{\mathbf{x}}_j(k|k)$ equals the observed measurements and velocity zero (one point initialization), and the initial position standard deviation (s.d.) is 0.3b (px) in each component and the initial velocity s.d. is 3b (px/s) where b is the BB width.

For the ILS search of MLE4, the initial 4D camera parameter vector is set as $[0\ 1\ 0\ 0]'$ in equation (5). Based on the 2D assignment algorithm, the optimal 4D parameter vector is estimated by the ILS algorithm (see, e,g., [2]).

### B. Small Number of Targets—HTX Video, Assignment With Exhaustive Search

In this section, we examine the scenario considered in [25] with a small number of targets.[7] The UAS only changes its position at a fixed altitude during recording. The sampling frequency is 30 Hz (30 frames/s). The frame has a size of $1920 \times 1080$ px. The width of the BBs is around 15 px and the height of the BBs is around 42 px. The average velocity of the BBs is around 23 px/s. Thus, we choose the process noise power spectral density as[8] $q = 16$ (px$^2$/s$^3$) and the measurement noise covariance matrix as $R = \text{diag}[9\ 9]$ (px$^2$).

The performance of tracking accuracy is evaluated by the *average track residual errors*—average track residual errors (ATRE) at each frame, as follows:

$$e(k) = \frac{1}{N_t(k)} \sum_{j=1}^{N_t(k)} \sqrt{\tilde{x}(k)_{i(j)}^2 + \tilde{y}(k)_{i(j)}^2}. \tag{34}$$

where $\tilde{x}$ and $\tilde{y}$ are innovation (residual) errors (the difference between the corrected predicted positions and

[7]https://github.com/zijiaoTian58/HTX_Dataset.
[8]The root mean square (RMS) change in the velocity over a sampling interval $T$ is $\sqrt{qT}$, which for $T = 1/30$ s, yields 0.7 px/s.

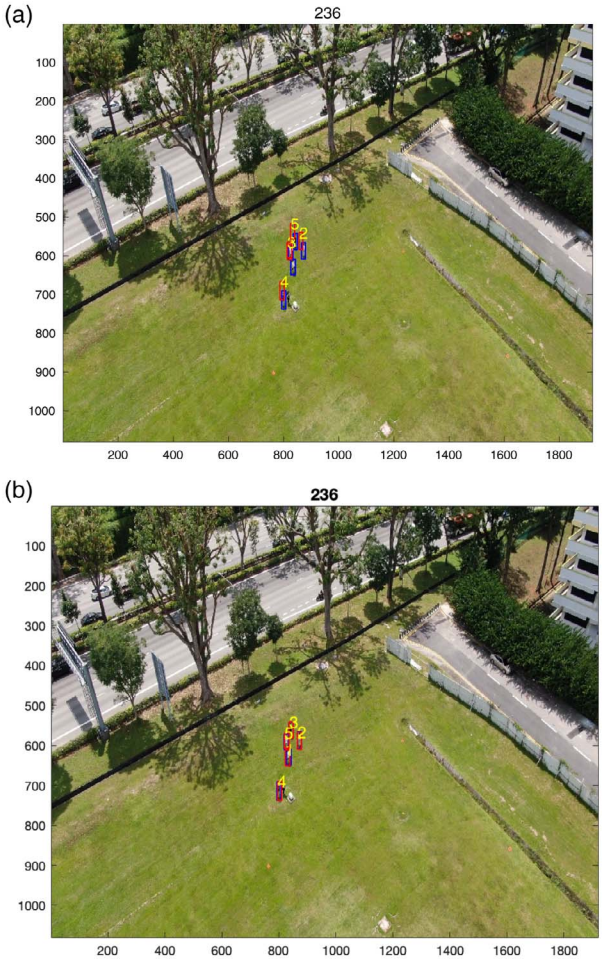



Figure 4. HTX scenario: tracking results at frame 236 based on (a) the GMIC method and (b) the proposed method (MLE4). Blue bounding boxes are the actual measurements, and the red bounding boxes are the tracking results. (a) Tracking based on the GMIC method. (b) Tracking based on MLE4.

the measured position based on the most recent assignment) of each target.

The position residual errors of the GMIC (gating) versus MLE3 are illustrated in Fig. 3. The camera adjusts its FoV to track targets, resulting in camera motion changes around frames 70 and 150–300. Our proposed MLE3 (assignment with exhaustive search) yields much smaller position residual errors than GMIC without camera motion parameters estimation. The ATRE for GMIC and MLE3, MLE4 are 9.04, 2.94, and 2.48, respectively. MLE4 has ATRE about 73% smaller than the GMIC method. The results demonstrate the robustness of the proposed methods. MLE3 and MLE4 have the ability to maintain tracking accuracy even in dynamic camera scenarios, while GMIC's reliance on validation gates can lead to incorrect association and degraded tracking in the presence of sudden camera movements.

Figure 4 shows the tracking results (GMIC versus MLE4) at frame 236. The blue BBs represent actual measurements, and red BBs indicate tracking results. For

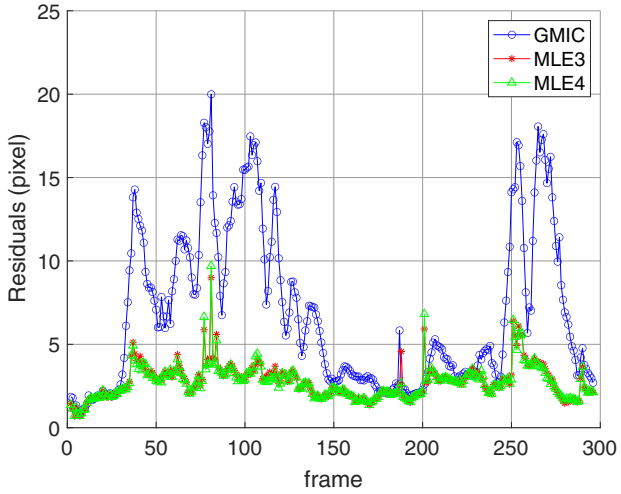



Figure 5. VDD scenario 1: Average track residual errors for GMIC, MLE3, and MLE4 are 6.93, 2.77, and 2.72, respectively.

MLE4, the red BBs for all targets almost cover the blue BBs, whereas for GMIC, the red BBs have large deviations from the blue BBs. This indicates that the proposed MLE4 performs better tracking than the GMIC when the camera has motion change at frame 236.

### C. Large Number of Targets—VDD Video, Assignment With Hungarian

Two scenarios with multiple targets from VisDrone Dataset (VDD)[30][9] are considered. The sampling frequency is 15 Hz (15 frames/s), and the frame has a size of 1344 $\times$ 756 px. The UAS can change its position and altitude and thus result in more complex camera motion change.

For Scenario 1 (VDD #0000088_00290), there are more than 60 targets in one frame. The average size (width and height) of the BBs is around [35 80] px. The average velocity of the BBs is around [5 45] px/s. The process noise power spectral density is chosen as[10] $q = 49$ (px$^2$/s$^3$) and the measurement noise covariance matrix as $R = \text{diag}[9\ 9]$ (px$^2$).

As shown in Fig. 5, the position residual errors of the GMIC method are significantly larger compared to those obtained with MLE3 method and MLE4 method (assignment with Hungarian), especially when the camera is panning or tilting during frames 80–140 and 250–270. The average track residual errors during the whole frames for GMIC, MLE3 and MLE4 are 6.93, 2.77, and 2.72, respectively. The tracking results (GMIC versus MLE4) at frame 105 are shown in Fig. 6.

For Scenario 2 (VDD #0000099_02109), the average size (width and height) of the BBs is around [20 50] px. The average velocity of the BBs is around [4 40] px/s.

[9]https://github.com/VisDrone/VisDrone-Dataset.
[10]The RMS change in the velocity over a sampling interval $T$ is $\sqrt{qT}$, which for $T = 1/15$ s, yields 1.8 px/s.

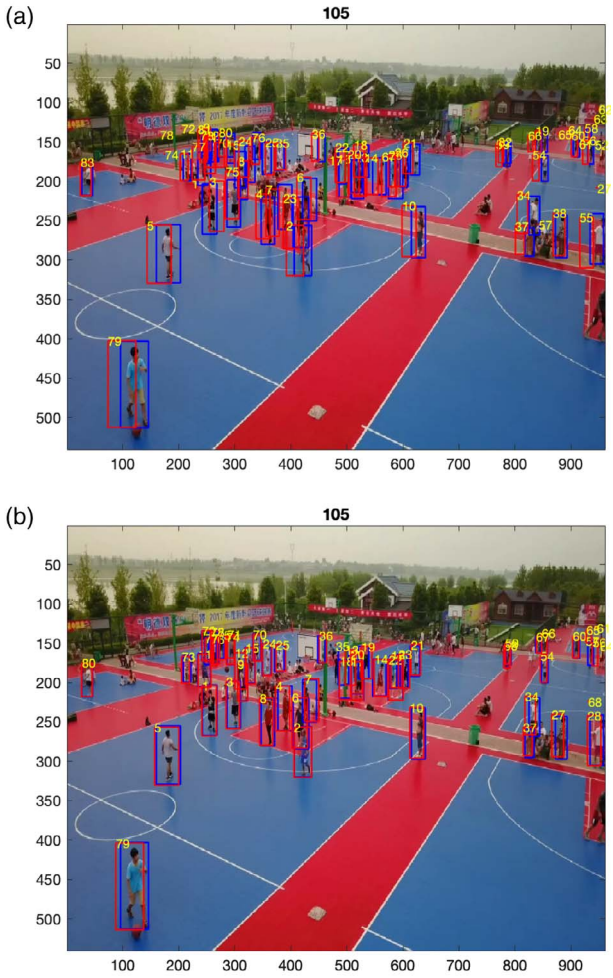



Figure 6. VDD scenario 1: tracking results at frame 105 based on (a) the GMIC method and (b) the proposed method (MLE4). (a) Tracking based on the GMIC method. (b) Tracking based on MLE4.

Thus, we choose the process noise power spectral density as[11] $q = 16$ (px$^2$/s$^3$) and the measurement noise covariance matrix as $R = \text{diag}[9\ 9]$ (px$^2$).

Similarly, our proposed methods (MLE3 and MLE4) show smaller position residual errors than the GMIC method (ATRE = 6.47, 2.31, 2.12), as seen in Fig. 7, especially around frame 600. Note that the MLE4 outperforms MLE3. This is because the camera increases its altitude during frames 500–650 and has slight roll. The tracking results at frame 570 are shown in Fig. 8, revealing that MLE4 significantly surpasses GMIC in tracking accuracy.

Next, we evaluate in detail the tracking quality for Scenario 2. There are two metrics to evaluate the tracking performance: the number of track breaks and the number of track swaps. As shown in Table II, the proposed MLE methods outperform the GMIC method, yielding fewer track breaks and track swaps. Fig. 9 shows the track swaps and track breaks when the camera has

[11]This corresponds to an RMS change in the velocity over $T = 1/15$ s of 1 px/s.

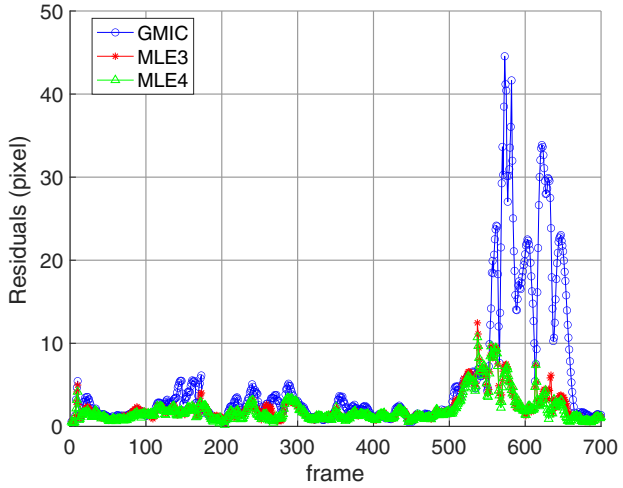



Figure 7. VDD scenario 2: Average track residuals errors for GMIC, MLE3, and MLE4 are 6.47, 2.31, and 2.12, respectively.

abrupt movements. Track breaks refer to interruptions in the continuous tracking of a target. Track swaps refer to the instances where the identity of a track is mistakenly switched between two different targets. The top three figures show that there are track IDs mistakenly swapped by the GMIC method, whereas the bottom figures show no track swaps by the MLE4. Due to the obstruction of buildings (yellow pavilion), track 13 from the top figures is broken, while MLE4 continues track 13 without track breakage or swap.

## VI. CONCLUSIONS

In this paper, we carried out M2TA with camera motion parameters estimation for drone-captured video to reduce the effect of sudden camera movement. The camera parameter vector is extended from 3D (pan, tilt, and zoom) to 4D (pan, tilt, zoom, and roll). Based on the estimation, the proposed approach not only yields improved M2TA pairings but also can provide better state estimation in the update step. The real data results show that the proposed approach (MLE4 and MLE3) can reduce the effects of sudden camera movement. MLE4 is better than MLE3. The robustness has been confirmed by using the algorithm in diverse situations with good results. The tracking accuracy and tracking quality are much better than the GMIC (gating) method.

Table II
Tracking Quality for Scenario 2

| | # Track Breaks | # Track Swaps |
|---|---|---|
| GMIC | 5 | 5 |
| MLE3 | 3 | 1 |
| MLE4 | 2 | 1 |

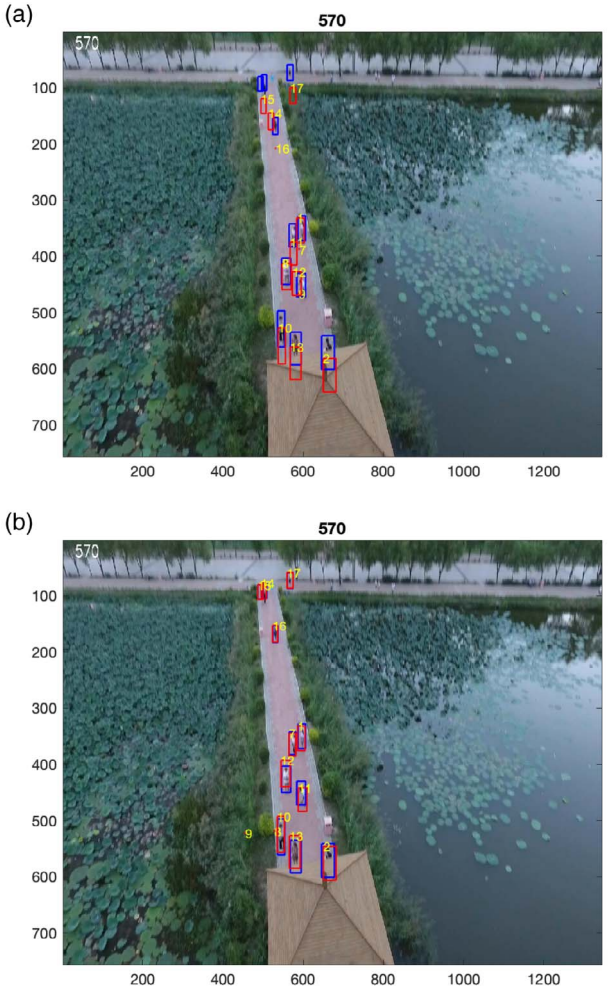



Figure 8. VDD scenario 2: tracking results at frame 570 based on (a) the GMIC method and (b) the proposed method (MLE4). (a) Tracking based on the GMIC method. (b) Tracking based on MLE4.

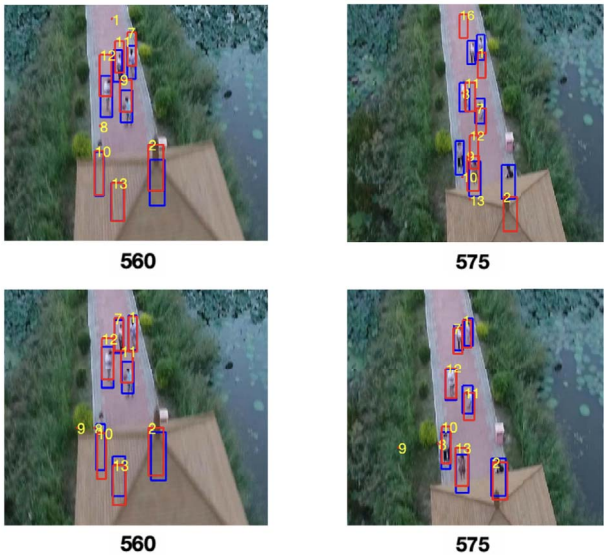



Figure 9. Track breaks and track swaps. Top figures: Tracks 9, 12, and 13 are mistakenly swapped by the GMIC method from frame 560 to frame 575 when the camera has abrupt movements. Bottom figures: There are no track swaps by the MLE4.

## APPENDIX A

The LLS method is used to solve for $\hat{\boldsymbol{\zeta}}$ directly in the MLE3 method. For an assignment with $\{i \leftrightarrow j\}_{i=1}^{N}$ ($N$ is the number of M2TA pairings), the cost based on (4) is expressed as

$$c[i,\, j(i),\, k,\, \boldsymbol{\zeta}(k)] = ||\mathbf{z}_i(k) - \hat{\mathbf{z}}^{\kappa}_{j(i)}(k|k-1)||^2 \tag{35}$$

$$= [x_i(k) - \hat{x}_{j(i)}(k|k-1)\phi(k) - x_c(k)]^2 + [y_i(k) - \hat{y}_{j(i)}(k|k-1)\phi(k) - y_c(k)]^2 \tag{36}$$

where $j(i)$ denotes the track paired with measurement $i$.[12] Define the following stacked matrices:

$$\mathbf{H} = [\mathbf{H}_1 \ldots \mathbf{H}_N]', \tag{37}$$

$$\mathbf{v} = [\mathbf{v}_1 \ldots \mathbf{v}_N]', \tag{38}$$

$$\mathbf{R} = \text{diag}[\Re_1 \ldots \Re_N], \tag{39}$$

where

$$\mathbf{H}_j = \begin{bmatrix} \hat{x}_j & 1 & 0 \\ \hat{y}_j & 0 & 1 \end{bmatrix}, \tag{40}$$

$$\mathbf{v}_j = \begin{bmatrix} x_{i(j)} - \hat{x}_j \\ y_{i(j)} - \hat{y}_j \end{bmatrix}, \tag{41}$$

$$\Re_j = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}, \tag{42}$$

with the estimate of the innovation variance (for statistical significance, to be discussed later) is

$$\hat{\sigma}^2 = \frac{1}{2N - n_\zeta}(\mathbf{H}\hat{\boldsymbol{\zeta}} - \mathbf{v})'(\mathbf{H}\hat{\boldsymbol{\zeta}} - \mathbf{v}), \tag{43}$$

where $i(j)$ in (41) is the index of the measurement associated with track $j$, $j = 1, \ldots, N$ and $n_\zeta$ is the number of camera parameters.

Then the LLS problem for pairs $i(j)$, $j$ is given by

$$\arg\min_{\zeta} \sum_{j=1}^{N} (\mathbf{H}_j\boldsymbol{\zeta} - \mathbf{v}_j)' \Re_j^{-1} (\mathbf{H}_j\boldsymbol{\zeta} - \mathbf{v}_j), \tag{44}$$

or, without the summation (with the stacked matrices)

$$\arg\min_{\zeta} (\mathbf{H}\boldsymbol{\zeta} - \mathbf{v})' \mathbf{R}^{-1} (\mathbf{H}\boldsymbol{\zeta} - \mathbf{v}), \tag{45}$$

Finally, $\hat{\boldsymbol{\zeta}}$ is obtained by minimizing the quadratic error (44),

$$\hat{\boldsymbol{\zeta}} = (\mathbf{H}'\mathbf{R}^{-1}\mathbf{H})^{-1}\mathbf{H}'\mathbf{R}^{-1}\mathbf{v} = \left(\sum_{j=1}^{N} \mathbf{H}_j'\mathbf{H}_j\right)^{-1} \mathbf{H}'\mathbf{v}, \tag{46}$$

with its covariance matrix given by

$$\mathbf{P}_{\hat{\zeta}} = (\mathbf{H}'\mathbf{R}^{-1}\mathbf{H})^{-1} = \left(\sum_{j=1}^{N} \mathbf{H}_j'\Re_j^{-1}\mathbf{H}_j\right)^{-1}. \tag{47}$$

[12] The pairing notations $i(j)$ and $j(i)$ are equivalent.

## APPENDIX B

The partial derivatives equation (20) in Section III.B are as follows:

$$\frac{\partial h_1}{\partial \rho} = (-\hat{x}_{i(j)}(k|k-1)\sin\rho(k) + \hat{y}_{i(j)}(k|k-1)\cos\rho(k))\phi(k), \tag{48}$$

$$\frac{\partial h_1}{\partial \phi} = \hat{x}_{i(j)}(k|k-1)\cos\rho(k) + \hat{y}_{i(j)}(k|k-1)\sin\rho(k), \tag{49}$$

$$\frac{\partial h_1}{\partial x_c} = 1, \tag{50}$$

$$\frac{\partial h_1}{\partial y_c} = 0, \tag{51}$$

$$\frac{\partial h_2}{\partial \rho} = (-\hat{y}_{i(j)}(k|k-1)\sin\rho(k) - \hat{x}_{i(j)}(k|k-1)\cos\rho(k))\phi(k), \tag{52}$$

$$\frac{\partial h_2}{\partial \phi} = (\hat{y}_{i(j)}(k|k-1)\cos\rho(k) - \hat{x}_{i(j)}(k|k-1)\sin\rho(k)), \tag{53}$$

$$\frac{\partial h_2}{\partial x_c} = 0, \tag{54}$$

$$\frac{\partial h_2}{\partial y_c} = 1. \tag{55}$$

## REFERENCES


[1] G. Allebosch, D. V. Hamme, P. Veelaert, and W. Philips "Robust pan/tilt compensation for foreground–background segmentation," *Sensors*, vol. 19, no. 12, p. 2668, 2019.

[2] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan *Estimation With Applications to Tracking and Navigation*, Hoboken. NJ, USA: Wiley, 2001.

[3] Y. Bar-Shalom, P. Willett, and X. Tian *Tracking and Data Fusion: A Handbook of Algorithms*. YBS Publishing, Storrs, CT, USA, 2011.

[4] S. Blackman and R. Popoli *Design and Analysis of Modern Tracking Systems*, Dedham, MA, USA: Artech House, 1999.

[5] C. Cadena et al. "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.

[6] A. Censi, A. Fusiello, and V. Roberto "Image stabilization by feature tracking," *Proc. 10th Int. Conf. Image Anal. Process.*, Venice, Italy, 1999, pp. 665–667.

[7] M.-N. Chapel and T. Bouwmans "Moving objects detection with a moving camera: A comprehensive review," *Comput. Sci. Rev.*, vol. 38, p. 100310, 2020.

[8] W. Choi, C. Pantofaru, and S. Savarese "A general framework for tracking multiple people from a moving camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 7, pp. 1577–1591, Jul. 2013.

[9] J. Edmonds and R. Karp "Theoretical improvements in algorithmic efficiency for network flow problems," *J. ACM.* vol. 19, no. 2, pp. 248–264, Apr. 1972.

[10] S. Erturk "Digital image stabilization with sub-image phase correlation based global motion estimation," *Trans. Consum. Electron.*, vol. 49, no. 4, pp. 1320–1325, Nov. 2003.

[11] G. D. Evangelidis and E. Z. Psarakis "Parametric image alignment using enhanced correlation coefficient maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1858–1865, Oct. 2008.

[12] B. Kiefer et al. "1st workshop on maritime computer vision (MaCVi) 2023: Challenge results," *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2023, pp. 265–302.

[13] S. Li and D.-Y. Yeung "Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models," *AAAI*, vol. 31, no. 1, Feb. 2017.

[14] A. Litvin, J. Konrad, and W. Karl "Probabilistic video stabilization using Kalman filtering and mosaicking," *Proc. SPIE*, vol. 5022, pp. 663–674, May 2003.

[15] E. Mingkhwan and W. Khawsuk "Digital image stabilization technique for fixed camera on small size drone," in *Proc. 3rd Asian Conf. Defence Technol.* 2017, pp. 12–19.

[16] M. Mueller, G. Sharma, N. Smith, and B. Ghanem "Persistent aerial tracking system for UAVs," 2016 *IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Daejeon, Korea (South), 2016, pp. 1562–1569.

[17] M. Mueller, N. Smith, and B. Ghanem "A Benchmark and Simulator for UAV Tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 445–461.

[18] A. Ramachandran and A. K. Sangaiah "A review on object detection in unmanned aerial vehicle surveillance," *Int. J. Cogn. Comput. Eng.*, vol. 2, pp. 215–228, 2021.

[19] P. Rawat and J. Singhai "Review of motion estimation and video stabilization techniques for hand held mobile video," signal and image process: *An Int. J.*, vol. 2, no. 2, pp. 159–168, Jun. 2011.

[20] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2016, pp. 779–788.

[21] J. Redmon and A. Farhadi "Yolov3: An incremental improvement," 2018, *arXiv:1804.02767*.

[22] D. Reid "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. 24, no. 6, pp. 843–854, Dec. 1979.

[23] W. Sohn and N. D. Kehtarnavaz "Analysis of camera movement errors in vision-based vehicle tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 1, pp. 57–61, Jan. 1995.

[24] J. Thomas, J. Welde, G. Loianno, K. Daniilidis, and V. Kumar "Autonomous flight for detection, localization, and tracking of moving targets with a small quadrotor," *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1762–1769, Jul. 2017.

[25] Z. Tian, Y. Bar-Shalom, R. Yang, H. J. Huang, and G. W. Ng "Interframe association of YOLO bounding boxes in the presence of camera panning and zooming," in *Proc. 26th Int. Conf. Inf. Fusion*, 2023, pp. 1–7.

[26] B.-N. Vo et al. "Multitarget tracking," in *Wiley Encyclopedia of Electrical and Electronics Engineering*. New York, NY, USA: Wiley, 2015.

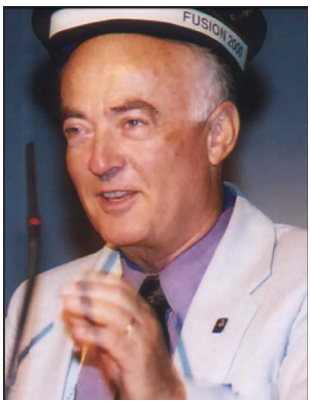
[27] S. Yeom and I.-J. Cho
"Detection and tracking of moving pedestrians with a small unmanned aerial vehicle," Appl. Sci., vol. 9, Art. no. 3359, 2019.

[28] J. Yang, D. Schonfeld, and M. Mohamed
"Robust video stabilization based on particle filter tracking of projected camera motion,"
*IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, pp. 945–954, Jul. 2009.

[29] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu
"Object detection with deep learning: A review,"
*IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.

[30] P. Zhu et al.
"Detection and tracking meet drones challenge,"
*IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 11, pp. 7380–7399, Nov. 2022.

**Zijiao Tian** received her B.Sc. degree in electrical engineering from Xidian University, Xi'an, Shaanxi, China, in 2019, M.Sc. and Ph.D. degrees in electrical engineering from the University of Connecticut (UConn), Storrs, CT, USA, in 2023 and 2024, respectively. She is currently a Research Scientist with Intelligent Fusion Technology, Inc (IFT), Germantown, MD, USA. Her research interests include statistical signal processing, target detection and tracking, sensor fusion, positioning, navigation, and timing (PNT). She was awarded the Summer Doctoral Dissertation Fellowship at UConn in 2024.




**Yaakov Bar-Shalom** (IEEE F'84) received the B.Sc. and M.Sc. degrees in electrical engineering from the Technion in 1963 and 1967, respectively, and the Ph.D. degree from Princeton University, Princeton, NJ, USA in 1970. He is currently a Board of Trustees Distinguished Professor with the ECE Department and Marianne E. Klewin Professor with the University of Connecticut. His current research interests are in estimation theory, target tracking, and data fusion. He has published over more than 650 papers and book chapters (more than 71 000 citations, $h = 103$). He coauthored/edited eight books, including *Tracking and Data Fusion* (YBS Publishing, 2011). He has been elected Fellow of IEEE for "contributions to the theory of stochastic systems and of multitarget tracking". He served as an Associate Editor for the Transactions on Automatic Control and Automatica. He was General Chairman of the 1985 ACC, General Chairman of FUSION 2000, President of ISIF in 2000 and 2002, and Vice President for Publications from 2004 to 2013. Since 1995, he has been a Distinguished Lecturer of the IEEE AESS. He is a corecipient of the M. Barry Carlton Award for the best paper in the IEEE TAE Systems in 1995 and 2000. In 2002, he received the J. Mignona Data Fusion Award from the DoD JDL Data Fusion Group. He is a member of the Connecticut Academy of Science and Engineering. In 2008, he was awarded the IEEE Dennis J. Picard Medal for Radar Technologies and Applications, and in 2012, the Connecticut Medal of Technology. He has been listed by academic.research.microsoft (top authors in engineering) as #1 among the researchers in aerospace engineering based on the citations of his work. He is the recipient of the 2015 ISIF Award for a Lifetime of Excellence in Information Fusion. This award has been renamed in 2016 as the ISIF Yaakov Bar-Shalom Award for a Lifetime of Excellence in Information Fusion. He has the following Wikipedia page: https://en.wikipedia.org/wiki/Yaakov Bar-Shalom. He is also the recipient (with H. Blom) of the IEEE AESS Pioneer Award for the invention of the IMM Estimator. He has been listed in the Stanford Top 2% Researchers List in 2023. He is also co-recipient of the 2023 Naval Research Lab Alan Berman Research Publication Award.

**Rong Yang** received her B.E. degree in information and control from Xi'an Jiao Tong University, Xi'an, Shaanxi, China, in 1986, M.Sc. degree in electrical engineering from National University of Singapore, Singapore, in 2000, and Ph.D. degree in electrical engineering from Nanyang Technological University, Singapore, in 2012. She is currently a Principal Member of Technical Staff at DSO National Laboratories, Singapore. Her research interests include passive tracking, low observable target tracking, GMTI tracking, hybrid dynamic estimation and data fusion. She was the Publicity and Publication Chair of FUSION 2012 and received the FUSION 2014 Best Paper Award (First Runner-Up).

**Huang Hong'An Jack** was born in Singapore in 1983. He received the B.E. degree from National University of Singapore (NUS), Singapore, in 2008. He is currently a Senior Member of Technical Staff with DSO National Laboratories, Singapore. His research interests include target tracking, including GMTI tracking, passive tracking, and image tracking. He received the FUSION 2014 Best Paper Award (First Runner-Up).

**Gee Wah Ng** received the M.Sc. and Ph.D. degrees from the University of Manchester Institute of Science and Technology, Manchester, UK. He is currently a Distinguished Member of Technical Staff at DSO National Laboratories, Singapore, and a Director at Home Team Science and Technology Agency (HTX), Singapore. He has delivered many projects in the decision support areas and has authored three books. He is active in international conferences in the areas of information fusion and intelligent systems. His research interests in data and information fusion include target tracking, computational intelligence, machine learning, self-tuning, and sensor networks. He was the Program Chair of FUSION 2012 and received the FUSION 2014 Best Paper Award (First Runner-Up).