3D Tracking of Mating Events in Wild Swarms of the Malaria Mosquito *Anopheles gambiae*

Sachit Butail[1], Nicholas Manoukis[4], Moussa Diallo[3], Alpha S. Yaro[3], Adama Dao[3], Sekou F. Traoré[3], José M. Ribeiro[2], Tovi Lehmann[2] and Derek A. Paley[1]

***Abstract*— We describe an automated tracking system that allows us to reconstruct the 3D kinematics of individual mosquitoes in swarms of *Anopheles gambiae*. The inputs to the tracking system are video streams recorded from a stereo camera system. The tracker uses a two-pass procedure to automatically localize and track mosquitoes within the swarm. A human-in-the-loop step verifies the estimates and connects broken tracks. The tracker performance is illustrated using footage of mating events filmed in Mali in August 2010.**

I. INTRODUCTION

Observation of wild mosquito swarms present an opportunity to study mating behavior [1], [2]. In swarms of *An. gambiae* the female enters a swarm composed almost entirely of males [1]. A three-dimensional reconstruction of the movement of individual mosquitoes and mating couples can provide a deeper insight into the basis of mate selection. Once automated, the procedure can generate large datasets that enable application of statistical analyses developed for other animal aggregations [3], [4], [5]. Such analyses can form the first steps towards strategies of vector control.

Aided by advances in computer vision, high-resolution and high frame-rate cameras have been used to observe and quantify position of flying insects in the laboratory [6], [7], [8], [9], as well as in the field [10]. In the case of mosquito swarms, previous research has focused on reconstructing the position of individual mosquitoes within a swarm at periodic time intervals [2], [11]. Each mosquito is an identity-less point in space and the analyses have mainly focused on swarm density and structure [2]. The task of connecting the points between multiple views and through time is nontrivial—one that requires automation and accuracy. Tracking wild mosquitoes poses challenges such as low lighting, a cluttered background, a variable number of targets, and a boundary-less environment. A multi-camera setup with a large baseline [9] reconstructs positions accurately, but is difficult to implement in the field.

Challenges to tracking mosquitoes and other insects include: (1) the three-dimensional position of each mosquito is not directly observable; (2) three-dimensional reconstruction of a point object using a single camera is not possible, so at least two cameras must be used; and (3) the location measurements in pixels are subject to noise, which introduces uncertainty in the estimates.

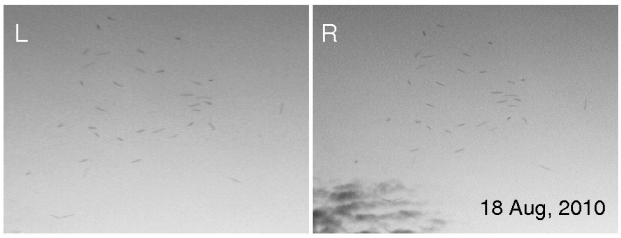

Fig. 1. The pair of images above are magnified and enhanced versions of raw footage obtained from field experiments in Mali.

In a multi-target, multi-camera system we must also address data association, which entails assigning measurements to targets across multiple views and across time. In [10] more than a hundred bats are tracked using a Kalman filter to process each camera view, in conjunction with a multi-dimensional assignment strategy. Three-dimensional tracks are reconstructed from individual views. In [7], a swarm of fruitflies is tracked in an acrylic box frame by setting up the problem of data association across views and across time in the form of a global optimization problem that is solved at every step. In [9] multiple fruit flies are tracked in real time in a laboratory using an extended Kalman filter and nearest-neighbor data association.

In this paper we describe an automated tracking system that allows us to reconstruct the three-dimensional kinematics of individual mosquitoes in wild swarms of *Anopheles gambiae* filmed in a rural village in Mali in August 2010. We film from two cameras synchronously at 25 frames per second. (The frame rate is limited by the ambient light.) The mosquitoes move in an unpredictable fashion and appear as streaks when not moving towards or away from the camera. At high speeds, the mosquito streaks fade, making them hard to detect. Because the swarms are dense, occlusions are frequent, and appear often in both camera frames. Fig. 1 shows a pair of magnified and enhanced sample frames.

The contributions of the paper are (1) we provide a method to probabilistically extract mosquito velocity information from streaks on the camera image plane; (2) we isolate faded streaks by altering the segmentation threshold in a specific region; and (3) using the expected orientation of streaks, we resolve occlusions on the image plane into individual mosquitoes. Collectively, these contributions enable us to apply tools from computer vision and estimation theory to

[1]Department of Aerospace Engineering, University of Maryland, College Park, MD 20742, USA {sbutail,dpaley}@umd.edu

[2]Laboratory of Malaria and Vector Research, National Institute of Allergy and Infectious Diseases, NIH, Bethesda, MD 20892, USA

[3]Malaria Research and Training Center, Faculté de Médecine, de Pharmacie et d'Odontostomatologie, Université de Bamako, Mali

[4] US Pacific Basin Agricultural Research Center, Agricultural Research Service, U.S. Department of Agriculture, Hilo, HI 96720, USA

implement a tracking framework that yields unprecedented data on mating events in malarial mosquitoes.

The paper is organized as follows: Section II provides a background on image-processing, nonlinear estimation, and data association. Section III presents the likelihood function that extracts velocity information from streaks on the image. We also describe a novel procedure to resolve occlusions by modelling an image blob as a combination of streaks. Section IV describes the experimental results. Section V summarizes the paper and provides a description of ongoing work.

II. Multi-target tracking

Our objective fits within the general framework of multi-target tracking using multiple cameras. We use a stereo-camera setup to film mosquito swarms in the field and a novel tracking framework to process the footage offline. There are three components of our tracking framework: image processing, nonlinear estimation, and data association.

Image processing: In situ observation of mosquito swarms prevents us from developing an initial background model to segment the mosquitoes out of the image stream. A dynamic background can be created by choosing the highest intensity point within a sliding window [12]. At time step k

$$B_{u,v}[k] = \max_{i \in [k-d, k+d]} B_{u,v}[i], \tag{1}$$

where $B_{u,v}$ is the background image at the pixel position (u, v) and $2d$ is the width of the sliding window. The foreground F is obtained by subtracting the background B from the current image I and applying a threshold t:

$$F_{u,v}[k] = \max(I_{u,v}[k] - B_{u,v}[k], t_{u,v}). \tag{2}$$

Noise and large insects are removed by applying a threshold on the area of the blobs obtained from segmentation. Due to large variation in movement patterns, we are still faced with faded streaks and missing observations. We address this problem by adaptively changing the thresholds on a section of the image as discussed in Section III.

Nonlinear estimation: The state of a mosquito is described at time k by a vector $\boldsymbol{X}[k] \in \mathbb{R}^9$, consisting of the three-dimensional position, velocity and acceleration. Measurements are denoted by a vector $\boldsymbol{Z}[k]$ that consists of a two-dimensional position in pixels in the image plane. The mapping from a three-dimensional position to a two-dimensional image on the camera is nonlinear and requires techniques from nonlinear estimation. The posterior probability density function (pdf) is a conditional probability $P(\boldsymbol{X}|\boldsymbol{Z}^k)$ of the state estimate $\boldsymbol{X}$ given the measurements up to k, $\boldsymbol{Z}^k$. Recursive Bayesian estimation uses the measurement(s) at each step to maximize the posterior pdf. The state $\boldsymbol{X}[k+1]$ and measurements $\boldsymbol{Z}[k+1]$ are related to the state $\boldsymbol{X}[k]$ according to

$$\begin{aligned} \boldsymbol{X}[k+1] &= \boldsymbol{F}(\boldsymbol{X}[k], \boldsymbol{w}[k+1]) \\ \boldsymbol{Z}[k+1] &= \boldsymbol{H}(\boldsymbol{X}[k+1], \boldsymbol{n}[k+1]), \end{aligned} \tag{3}$$

where $\boldsymbol{F}$ represents the motion model, $\boldsymbol{H}$ represents the measurement model, and $\boldsymbol{w}$ and $\boldsymbol{n}$ are the instantaneous disturbance- and measurement-noise values.

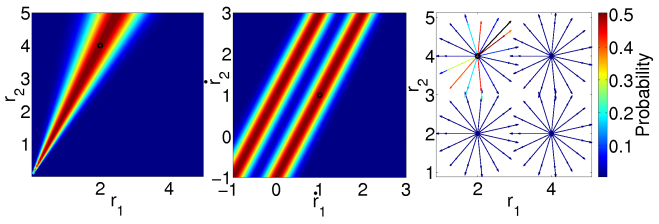

Fig. 2. Position (left), velocity (middle), and combined (right) likelihood functions on a plane parallel to the camera axis.

In a nonlinear estimator we use a likelihood function to represent the measurement model. (A likelihood function is the conditional probability $P(\boldsymbol{Z}|\boldsymbol{X})$ of a measurement given a target state.) Examples of nonlinear estimators include the extended Kalman filter, the unscented Kalman filter and the particle filter [13]. In this paper we use a particle filter to estimate the position of each mosquito in the swarm.

Data association: Common to all multi-target tracking systems is the task of associating measurements to targets, which entails maintaining the same measurement-target matches through consecutive frames and across camera views. In an environment with clutter it is typical to obtain more measurements than the number of targets. A simple strategy is to assign a measurement to the nearest measurement estimate; this strategy is called the nearest-neighbor filter [14]. An optimal Bayesian filter in this scenario takes into account all of the past history of measurement-target associations and branches out a path from each such pairing to assign a probability to the latest set of measurements. The number of paths in such a scenario increases exponentially with the number of measurements [15]. Strategies to speed up the process include pruning the paths [15] and using only a fixed number of previous associations [14].

III. Reconstructing mosquito kinematics

This section describes the likelihood function and adaptive thresholding used to segment individual mosquitoes from the background and reconstruct their 3D movement.

Likelihood function: The position $\boldsymbol{u}^c \in \mathbb{R}^2$ of the centroid of a blob in camera c is a function of the mosquito position $\boldsymbol{r}$ and camera focal length f^c: $\boldsymbol{u}^c(\boldsymbol{r}) = f^c \begin{bmatrix} r_1/r_3 & r_2/r_3 \end{bmatrix}^T$. Probabilistically, we represent the likelihood of $\boldsymbol{u}^c(\boldsymbol{r})$ as

$$P^c_{pos}(\boldsymbol{u}^c(\boldsymbol{r})|\boldsymbol{r}) = \mathbb{N}(\boldsymbol{u}^c(\boldsymbol{r}); \boldsymbol{u}^c, \Sigma_{pos}), \tag{4}$$

where $\mathbb{N}(\boldsymbol{u}(\boldsymbol{r}); \boldsymbol{u}, \Sigma)$ denotes a normal distribution function with mean $\boldsymbol{u}$ and noise covariance matrix Σ.

The trajectory of a mosquito during the exposure appears on the image plane in the form of a streak. Velocity information is extracted from a streak by assuming that a mosquito travels at constant velocity during the exposure. Given the velocity, $\dot{\boldsymbol{r}} = \begin{bmatrix} \dot{r}_1 & \dot{r}_2 & \dot{r}_3 \end{bmatrix}^T$, and the time of exposure t_e, the start and end points during exposure with $\boldsymbol{r}$ as the midpoint are given by $\left[\boldsymbol{r} - \dot{\boldsymbol{r}}\frac{t_e}{2}, \boldsymbol{r} + \dot{\boldsymbol{r}}\frac{t_e}{2}\right]$. Projecting these

points on camera c we get image plane velocity $\boldsymbol{v}^c \in \mathbb{R}^2$

$$\boldsymbol{v}^c = f^c \begin{bmatrix} \frac{\dot{r}_1 r_3 - r_1 \dot{r}_3}{r_3^2 - \dot{r}_3^2 t_e^2/4} \\ \frac{\dot{r}_2 r_3 - r_2 \dot{r}_3}{r_3^2 - \dot{r}_3^2 t_e^2/4} \end{bmatrix}. \tag{5}$$

Since the streak itself does not provide a sense of direction there is a forward-backward ambiguity that is represented by computing the likelihood function of both positive and negative values of the velocity. Using (5) the likelihood function can be written as

$$\begin{aligned} P^c_{vel} =& P(\boldsymbol{v}^c|\boldsymbol{r},\dot{\boldsymbol{r}}) + P(-\boldsymbol{v}^c|\boldsymbol{r},\dot{\boldsymbol{r}}) \\ =& \mathbb{N}(\boldsymbol{v}^c; \boldsymbol{v}^c(\boldsymbol{r},\dot{\boldsymbol{r}}), \Sigma_{vel}) + \mathbb{N}(-\boldsymbol{v}^c; \boldsymbol{v}^c(\boldsymbol{r},\dot{\boldsymbol{r}}), \Sigma_{vel}). \end{aligned} \tag{6}$$

The combined position and velocity likelihood function is

$$P(\boldsymbol{Z}|\boldsymbol{X}) = \prod_{c=1}^{2} P^c_{pos} P^c_{vel}, \tag{7}$$

as depicted in Fig 2.

Adaptive thresholding: We use the probability of the nearest measurement assignment to determine if the region near the target estimate should be explored for faded streaks. We vary the background subtraction parameters: the threshold on intensity t; and the size of the sliding window d. For each frame, we project the estimate $\hat{\boldsymbol{X}}$ onto camera c to get $\hat{\boldsymbol{Z}}^c$. A gating volume generated as a chi-square distribution on innovation with two degrees of freedom [14] is searched for possible measurements. In the absence of a measurement, we change the intensity threshold followed by the size of sliding window until a measurement is found. The area threshold is maintained to avoid picking up noise. All measurements are then weighted based on (2) to get the best association. We have

$$\Delta t_{u,v} = t_{u,v}(K_c - 1), \tag{8}$$

where $0 < K_c < 1$. In conjunction with the change in threshold, the sliding window d is increased by 1.

Resolving occlusions: An occlusion is detected if a blob on the image plane is assigned to multiple targets on the basis of proximity. We compute an objective function based on candidate positions and velocities of the mosquitoes in an occlusion by projecting the streaks onto the image plane. The best fit is defined in terms of the orthogonal distance of all points on the blob to the streaks from occluding mosquitoes. This process is formalized as follows:

Let the targets involved in an occlusion O be indexed by $o = \{1, \ldots, N_o\}$, where N_o is the number of targets in the occlusion. Collectively, the state of all such targets is $\boldsymbol{X}_O = \begin{bmatrix} \boldsymbol{X}_1 & \boldsymbol{X}_2 & \cdots & \boldsymbol{X}_{N_o} \end{bmatrix}^T$. $\boldsymbol{X}_o$ is estimated by solving

$$\begin{aligned} \hat{\boldsymbol{X}_O} &= \underset{\boldsymbol{X}_O}{\operatorname{argmin}} \sum_i q_i^*, \text{ where} \\ q_i^* &= \min_s \|\boldsymbol{u}_i - \hat{\boldsymbol{u}}(s)\|. \end{aligned} \tag{9}$$

$\boldsymbol{u}_i$ is a point on the blob indexed by i and $\hat{\boldsymbol{u}}$ is the combination of projected streaks for all targets in the occlusion. The initial values for velocity and position are based most recent estimates. The tracking algorithm is given below.

Mosquito swarm tracking algorithm

Input: Sequence of synced images from a stereo camera setup, I_l, I_r, and camera calibration matrices P_l, P_r

Initialize: Find measurement pairs that satisfy the epipolar constraint [16] and compute 3D estimates

For each time step k:

1: *Associate measurements:* For each target estimate, compute the likelihood function (7) for each measurement within the gating volume. If no measurement is found, change the threshold until a measurement is detected or the minimum threshold is reached.
2: *Resolve occlusions:* If two targets associate to the same measurement, use (9) to update the target estimates.
3: *Update* the state estimate of each target not in an occlusion.
4: *Initiate/terminate:* For measurement pairs that satisfy the epipolar constraint and are not associated to any target, create a tentative target. Terminate an existing target if no measurement is found within a gating volume. Terminate duplicate tracks based on successive occlusions.
5: *Predict:* Use a white noise acceleration motion model [17] to predict the target state for the next time-step.

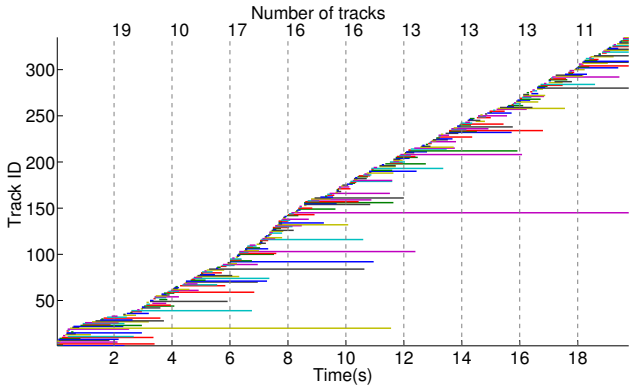

Fig. 3. Track integrity. The above plot shows the length and number of tracks generated from the automated tracker. The average (resp. max) actual swarm size for this sequence was 18 (resp. 21). The average (resp. max) track length was 21 frames (resp. 292). Vertical lines show the number of tracks at given frame. The tracks are linked by hand in post-processing.

After automated tracking, the tracks are displayed on a custom graphical user interface (GUI) developed in MATLAB that allows manual verification and linking of tracks. The final tracks are smoothed using a Kalman filter.

IV. Experimental Methods & Results

We used a pair of phase locked Hitachi KP-F120CL cameras in a stereo configuration. The video streams were recorded onto a 2.8 Ghz quad core laptop through an Imperx FrameLink Express frame grabber and Streampix software. Each camera was calibrated onsite using a checkerboard and the MATLAB Calibration Toolbox [18]. Reprojection error, which is a measure of calibration accuracy, was in sub-pixels for each camera. Relative camera orientation and position was determined by extrinsic calibration before filming. During filming, the camera azimuth and elevation were recorded to create a ground-fixed reference frame. Other environmental factors such as wind speed and direction and humidity were sampled once every 10 seconds.

We tested the automated tracker for position accuracy and identity maintenance by two methods: (1) quantifying the agreement on tracks of the same mosquito with two independent camera setups and (2) tracking an object with

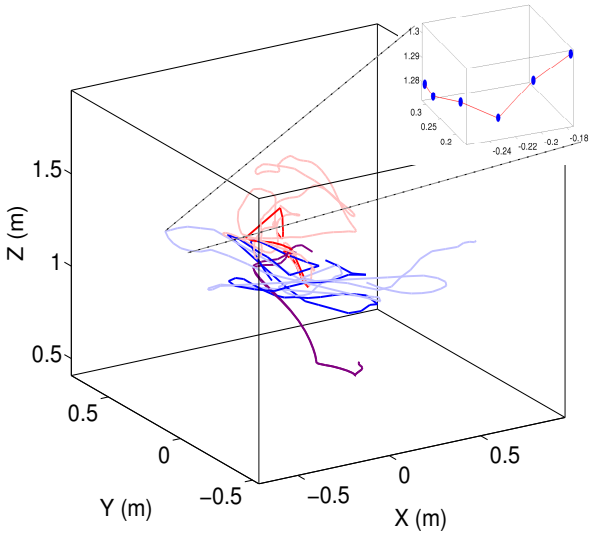

Fig. 4. Male-female tracks from a filming sequence in August 2010. The tracks show three different stages of coupling where the male and female mosquito are far apart (light blue and red), close (dark blue and red), couple (magenta). A section of the male track is magnified to show the 1σ error ellipsoids in three dimensions. See video of other tracking sequences at http://youtu.be/kuaMcVf501Y

known dimensions such as a checkerboard. We simultaneously tracked a single swarming event using two independent stereo camera rigs to validate accuracy in field. A common reference frame was created using the azimuth and elevation readings. The videos were time-synced between camera setups using a laser pointer. The mean distance between independent tracks of the same mosquito (200 data points) was 4.4 cm and the standard deviation was 1.3 cm. (Up to 3 cm error can be attributed to the inter-frame time difference between the camera systems.) Tracking a known target (a calibration checkerboard) yielded less than 2 cm error.

We also compared automated tracks with manually generated tracks. We randomly selected a mosquito swarming sequence and identified occlusions based on data association. Each of these occlusions were then presented to a human user who resolved them manually. We achieved 68 percent accuracy in resolving occlusions automatically as compared to 72 percent in tracking bats [10]. Track initiation and termination resulted in more tracks than the number of mosquitoes (Fig. 3); tracks are linked in post-processing. The average track length of 21 frames corresponds to 0.84 s as compared to 0.3 s in [7].

Tracking position and velocity of every mosquito in a swarm allows us to automatically detect mating events (Fig. 4). A mosquito couple flies in a distinct manner and speed. The speed of a target can be used in conjunction with the size and intensity of the blob in the image stream to automatically detect a slow, large couple.

V. Conclusion

In this paper we describe a tracking framework that is currently being used to reconstruct the three-dimensional trajectories of wild mosquito swarms. We use a probabilistic framework to recursively estimate the position of each mosquito and a robust data association method to find a mosquito if it is not segmented properly. The tracking system is a two-part process involving automated generation followed by human verification. Using position and velocity estimates allow us to automatically detect coupling events and analyze them. We are currently using this tracking system to generate high volumes of tracked data, which can be subject to statistical analyses for a deeper understanding of mosquito swarming and mating behavior.

VI. Acknowledgments

We gratefully acknowledge the travel support from the Laboratory of Malaria and Vector Research, National Institute of Allergy and Infectious Diseases.

References

[1] J. Charlwood and M. Jones, "Mating in the mosquito, Anopheles gambiae sl," *Physiological Entomology*, vol. 5, no. 4, pp. 315–320, 1980.

[2] N. C. Manoukis, A. Diabate, A. Abdoulaye, M. Diallo, A. Dao, A. S. Yaro, J. M. C. Ribeiro, and T. Lehmann, "Structure and Dynamics of Male Swarms of Anopheles gambiae," *J. Medical Entomology*, vol. 46, no. 2, pp. 227–235, 2009.

[3] M. Ballerini, N. Cabibbo, R. Candelier, A. Cavagna, E. Cisbani, I. Giardina, V. Lecomte, A. Orlandi, G. Parisi, A. Procaccini, and Others, "Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study," *Proc. Nat. Acad. Sciences*, vol. 105, no. 4, pp. 1232–1237, 2008.

[4] E. Wei, E. W. Justh, and P. S. Krishnaprasad, "Pursuit and an evolutionary game," *Proc. Royal Society A: Mathematical, Physical and Engineering Science*, vol. 465, no. 2105, pp. 1539–1559, 2009.

[5] A. Cavagna, A. Cimarelli, I. Giardina, G. Parisi, R. Santagati, F. Stefanini, and M. Viale, "Scale-free correlations in starling flocks." *Proc. Nat. Acad. of Sciences*, vol. 107, no. 26, pp. 11 865–11 870, 2010.

[6] S. Fry, M. Bichsel, P. Müller, and D. Robert, "Tracking of flying insects using pan-tilt cameras," *J. of Neuroscience Meth.*, vol. 101, no. 1, pp. 59–67, 2000.

[7] D. Zou, Q. Zhao, H. S. Wu, and Y. Q. Chen, "Reconstructing 3D motion trajectories of particle swarms by global correspondence selection," in *Proc. Int. Conf. Computer Vision*, 2009, pp. 1578–1585.

[8] J. Hardie and S. Young, "Aphid flight-track analysis in three dimensions using video techniques," *Physiological Entomology*, vol. 22, no. 2, pp. 116–122, 1997.

[9] A. D. Straw, K. Branson, T. R. Neumann, and M. H. Dickinson, "Multi-camera real-time three-dimensional tracking of multiple flying animals," *J. Royal Society Interface*, vol. 8, no. 56, pp. 395–409, 2010.

[10] Z. Wu, N. I. Hristov, T. L. Hedrick, T. H. Kunz, and M. Betke, "Tracking a large number of objects from multiple views," in *Int. Conf. on Computer Vision*. Citeseer, 2009, pp. 1546–1553.

[11] T. Ikawa, H. Okabe, T. Mori, K. Urabe, and T. Ikeshoji, "A method for reconstructing three-dimensional positions of swarming mosquitoes," *J. of Insect Behavior*, vol. 7, no. 2, pp. 237–248, 1994.

[12] A. Cavagna, I. Giardina, A. Orlandi, G. Parisi, A. Procaccini, M. Viale, and V. Zdravkovic, "The STARFLAG handbook on collective animal behaviour: 1. Empirical methods," *Animal Behaviour*, vol. 76, no. 1, pp. 217–236, 2008.

[13] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Proc.*, vol. 50, no. 2, pp. 174–188, 2002.

[14] Y. Bar-Shalom, *Tracking and data association*. San Diego, CA, USA: Academic Press Professional, Inc., 1987.

[15] D. Reid, "An Algorithm for Tracking Multiple Targets," *IEEE Trans. on Automatic Control*, vol. 24, no. 6, pp. 843–854, 1979.

[16] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[17] X. Rong Li and V. P. Jilkov, "Survey of maneuvering target tracking. Part I: Dynamic models," *IEEE Trans. Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

[18] J.-Y. Bouguet. Camera Calibration Toolbox for Matlab. [Online]. Available: http://www.vision.caltech.edu/bouguetj/calib_doc/index.html