Optimal fusion of video and RF data for detection and tracking with object occlusion

Benjamin Shapo*[a], Christopher Kreucher[b]
[a]Integrity Applications Incorporated, 15020 Conference Ctr. Dr. #100, Chantilly, VA USA 20151
[b]Integrity Applications Incorporated, 900 Victors Way #220, Ann Arbor, MI USA 48108

ABSTRACT

Occlusions can degrade object tracking performance in sensor imaging systems. This paper describes a robust approach to object tracking that fuses video frames with RF data in a Bayes-optimal way to overcome occlusion. We fuse data from these heterogeneous sensors, and show how our approach enables tracking when each modality cannot track individually. We provide the mathematical framework for our approach, details about sensor operation, and a description of a multisensor detection and tracking experiment that fuses real collected image data with radar data. Finally, we illustrate two benefits of fusion: improved track hold during occlusion and diminished error.

Keywords: Multisensor data fusion, Occlusion, Bayesian, Detection, Tracking, Particle Filtering

1. INTRODUCTION

The image processing community has recently shown great interest in algorithms for detecting and following objects in video. Investigators [3] have adopted many object representations for tracking purposes in video data. Tracking objects that have rich visual features often benefits from approaches like identifying primitive geometric shapes associated with the object or considering silhouette and color. For example, the work described in [4] describes finding feature points in the camera image sequence and jointly associating multiple features with existing tracks via the JPDAF algorithm [5]. In some applications, the object must be treated as a point target [39][40], and so features cannot be exploited for tracking.

A key requirement in many applications is performance in the presence of object occlusion. The literature addresses this need in several ways. The authors in [6] use video from a single camera to segment objects into individual features and track them separately. The algorithm assumes that some features will be visible and track those features. The approach in [7] accumulates grayscale features and overcomes occlusion using spatiotemporal context in the vicinity of the object. The algorithm in [8] determines correspondence between feature points in an image sequence. The authors deal with occlusions by applying their work to frames that are temporally separated. The authors in [9] integrate estimates from multiple features on different timescales, using features that include user silhouette, skin color, and face-pattern.

Other investigators address the occlusion problem by fusing data from multiple sensors. This approach is advantageous because it can provide more information than each individual sensor alone. Traditionally, multisensor fusion perform tracking at each sensor and then associate tracks across sensors [5], [12]-[13]. A common application is associating tracks across multiple sensors for improved localization [14]-[19]. Because hard tracking decisions are made at each sensor, this track-and-then-fuse method may not achieve optimal performance, but it may be necessary in systems that have limited processing or communication capabilities.

In the image processing community, multisensor fusion has focused almost exclusively on using multiple cameras. The approach in [26] uses multiple cameras to track features at each sensor. There are also multi-view results in [27], where the authors consider multiple looks at the tracked object from stationary and moving cameras. They track object appearance via a color-based representation and object motion with a Kalman Filter. The work described in [28] attacks a similar problem. The goal is to maintain track through occlusion and the approach includes object segmentation into features and tracking via a particle filter approach. [29] considers tracking a single object with multiple sensors, where each sensor performs local processing and shares data about the object at a central processor. This approach combines measurements via a likelihood approach and employs a particle filter to perform the tracking function. The work presented in [30] also adopts a Bayesian approach to fusing data from multiple sensors.

*bshapo@integrity-apps.com; phone 703.678.8672; fax 703.378.8978; www.integrity-apps.com

Signal Processing, Sensor/Information Fusion, and Target Recognition XXIII, edited by Ivan Kadar, Proc. of SPIE
Vol. 9091, 909106 · © 2014 SPIE · CCC code: 0277-786X/14/$18 · doi: 10.1117/12.2042781

The work described in this paper is based on the premise that fusing data from heterogeneous sensors can provide more robust detection and tracking performance than fusing data from homogeneous sensors. For example, video may be severely degraded at night or by rain, whereas RF is not affected by these conditions. Furthermore, rather than an approach which performs the tracking function at each sensor and then fuses the results, in this paper we advocate a fuse-before-track approach which performs data fusion before tracks are declared.

Other researchers have also investigated detection and tracking from the multi-sensor perspective by employing video sensors from different spectral bands [31]. Our work differs from this effort in several ways, including our use of RF to provide more sensing diversity and our fuse-before-track approach that differs from the JPDAF approach in [31].

This paper presents two main contributions. First, we show how the fuse-before-track approach naturally admits heterogeneous sensors and supports optimal Bayesian estimation. Second, we illustrate this utility with an experiment using a combination of an imaging sensor and a multistiatic RF array. The experiment includes severe sensor occlusion, generating a case where conventional track-and-then-fuse approaches fail.

The paper continues as follows. Section 2 gives a description of Bayesian filtering and its utility for multisensor data fusion. Section 3 provides results from a real data set, with truth available. Finally, Section 4 concludes the paper.

2. TECHNICAL APPROACH

Our approach is to fuse video image data with data from a multistatic RF array, for the purposes of detecting and tracking a moving object. Section 2.1 begins by outlining the Bayesian approach to detection and tracking. Next, Section 2.2 describes how the data fusion methodology, and its application in the case of occlusions. Section 2.3 continues by describing the occlusions and the behavior of our approach in their presence. Finally, Section 2.4 discusses the particular sensors involved in the experiment we performed.

2.1 Detection and Tracking Framework

Our approach to multitarget detection and tracking is based on the PDF ("Probability Density Function") tracker [21], [32]. The approach fuses heterogeneous sensor data through their likelihood function, allowing direct multisensor fusion. Its flexibility comes from the fact that its foundation is a rigorous mathematical framework and is also cast in a practical way that addresses real-world detection and tracking problems. The remainder of this sub-section describes the mathematical basis behind the PDF tracker.

In general, a tracker seeks to estimate the number of objects N_k at time k and the state of each x_n^k, $n = 1 \ldots N$. We are interested in tracking ground objects, so in this application the state is object position and velocity in two dimensions, $[x \ v_x \ y \ v_y]^T$. For the purpose of illustrating multi-modalilty fusion in the presence of occlusions, we assume a maximum of one object ($N_k \in \{0, 1\}$). The hypothesis H_0 that no object exists in the region and the hypothesis H_1 that one object exists span the target-detection space. The more general multitarget treatment is treated elsewhere [33].

The PDF tracker recursively computes $p(x_k, H_1^k \mid Z^k)$, which is the probability that an object exists and its state is x at time k given the observations Z^k. Z^k is the set of all measurements made up to and including the current time k, i.e., $Z^k = [z^0 \ z^1 \ \ldots \ z^k]$ and z^m is the (noise corrupted) measurement made at time m. The recursive nature of this approach is important in some applications because it provides object state estimates quickly. The estimated density provides details concerning object state. In particular, the tracker gives the estimated number of objects (in this application, 0 or 1), their locations, variance, and confidence.

Mathematically, we wish to estimate the hybrid continuous-discrete density $p(x_k, H_1^k \mid Z^k)$ for all k. This quantity can be separated into an object-present component and a object state estimate by multiple applications of the conditional probability law and Bayes' Theorem:

$$
\begin{aligned}
p(x^k, H_1^k | Z^k) &= \frac{p(x^k, H_1^k, Z^k)}{p(Z^k)} \\
&= \frac{p(x^k | H_1^k, Z^k) p(H_1^k, Z^k)}{p(Z^k)} \\
&= \frac{p(x^k | H_1^k, Z^k) p(H_1^k | Z^k) p(Z^k)}{p(Z^k)} \\
&= p(x^k | H_1^k, Z^k) p(H_1^k | Z^k) .
\end{aligned}
\tag{1}
$$

In this form, the expression is the product of the (discrete) object present probability $p(H_1^k \mid Z^k)$ and the (continuous) object state probability $p(x^k, H_1^k \mid Z^k)$. This approach separates the (still coupled) tasks of estimating the object present probability ("detection"), and estimating the object state density ("tracking").

In the real-time Bayesian approach, we assume an initial or prior estimate of the desired probabilities is present (perhaps completely uninformative, for example, uniform), and generate a recursive formula that relates the probabilities at each new time step to those from the previous time step. This formula updates the desired probability at each time step.

The discrete object present (detection) probability $p(H_1^k \mid Z^k)$ recursion integrates over all possible object-present states to obtain the result:

$$
\begin{aligned}
p(H_1^k | Z^k) &= \int p(x^k, H_1^k | Z^k) dx^k \\
&= p(H_1^k | Z^{k-1}) \frac{p(z^k | H_0^k)}{p(z^k | Z^{k-1})} \\
&\quad \times \int \frac{p(z^k | H_1^k, x^k)}{p(z^k | H_0^k)} p(x^k, H_1^k | Z^{k-1}) .
\end{aligned}
\tag{2}
$$

The continuous object state (tracking) probability updates by recursively folding in new measurements at time k:

$$
p(x^k | H_1^k, Z^k) = p(x^k | H_1^k, Z^{k-1}) \frac{p(z^k | H_1^k, x^k)}{p(z^k | H_0^k)} \frac{p(z^k | H_0^k)}{p(z^k | Z^{k-1}, H_1^k)} .
\tag{3}
$$

Therefore, the update for each x^k proceeds by predicting its probability forward according to a kinematic model and then updating according to the degree of agreement with new measurements via the likelihood ratio

$$
\Lambda(z^k | x^k) \equiv \frac{p(z^k | H_1^k, x^k)}{p(z^k | H_0^k)} .
\tag{4}
$$

Combining (3) and (4), we can then write

$$
p(x^k | H_1^k, Z^k) \propto p(x^k | H_1^k, Z^{k-1}) \Lambda(z^k | x^k) ,
\tag{5}
$$

where the constant of proportionality doesn't need to be explicitly calculated since the density must sum to 1. More detail on this approach is available in [2], with applications similar to the one described here in [29] and [30].

In our application, received data consists of pixels in the video stream and range/Doppler cells in the radar data. Since we use a point target model, the likelihood ratio expression for a sensor with C cells simplifies to

$$\frac{p(z^k|H_1^k,x^k)}{p(z^k|H_0^k)} = \frac{\prod_{c=1}^{C} p(z_c^k|H_1^k,x^k)}{\prod_{c=1}^{C} p(z_c^k|H_0^k)} \propto \frac{p(z_j^k|H_1^k,x^k)}{p(z_j^k|H_0^k)} \quad (6)$$

where j is the sensor pixel to which hypothesized target state x maps. In situations where we need to distinguish multiple sensors we will expand the notation to $p(z_{s,j}^k \mid H_0^k)$.

Both detection and tracking updates perform temporal evolution on their respective probability distributions. For the object present and absent probabilities $p(H_1^k \mid Z^{k-1})$ and $p(H_0^k \mid Z^{k-1})$, we use a mixing matrix approach. Similarly, a model on object kinematics $p(x^k \mid H_1^k, x^{k-1})$ performs the temporal update of the object state probability. This relationship is expressed in discrete time k as:

$$p(x^k|H_1^k,Z^{k-1}) = \int p(x^{k-1}|H_1^k,Z^{k-1})p(x^k|H_1^k,x^{k-1})dx^{k-1}. \quad (7)$$

2.2 Data Fusion Approach

Our fusion approach is based on the principle that it is better to perform thresholded decisions as late in the processing string as possible. We thus advocate a fuse-before-track approach to combining data from multiple sensors [20], which means that all sensor data is accumulated in a single tracker.

Fusing data from multiple sensors adds information and results in improved performance. Part of our approach's strength is that extension to multiple sensors is straightforward. For the case of two sensors, the new likelihood ratio becomes simply

$$\Lambda(z^k|x^k) = \frac{p(z_1^k|H_1^k,x^k)}{p(z_1^k|H_0^k)} \frac{p(z_2^k|H_1^k,x^k)}{p(z_2^k|H_0^k)}, \quad (8)$$

where z_1^k and z_2^k are the data from sensor #1 (e.g., a video sensor) and sensor #2 (e.g, an RF sensor), respectively, and vector measurement z^k refers to the current set of measurements from all sensors. We assume the data sources are conditionally independent given the object state. This assumption on the noise statistics is reasonable here, because the sensors are completely different modalities, and also because they are widely separated physically.

For P sensors, the expression extends to

$$\Lambda(z^k|x^k) = \prod_{p=1}^{P} \frac{p(z_p^k|H_1^k,x^k)}{p(z_p^k|H_0^k)} = \prod_{p=1}^{P} \Lambda_p(z^k|x^k) \quad (9)$$

This equation plugs into the Bayesian expression above in Equation (3) to allow the temporal recursion that describes the estimated object state over time. Note that x^k lacks a sensor index because we fuse the data from all sensors to obtain our fused object state estimate.

2.3 Data Fusion in the Presence of Occlusion

We model sensor occlusion as a complete lack of information from that sensor. We assume external factors, such as scene geometry (e.g., the object moving behind a known obstacle from the camera's perspective) or knowledge of other movers in the scene, determine the time steps at which occlusions occur. This model corresponds to the occluded sensor providing a flat (constant likelihood ratio for all hypothesized states x) update to the posterior density. In effect, it

results in an update that uses only data from non-occluded sensors.

Although not explored here, our framework allows more general occlusion modeling. For example, we can model occlusions that affect only certain target locations in the surveillance region by flatting the likelihood ratio in those areas. Models describing this effect must be sensor-specific.

2.4 Sensor Overview

Our experiments show how this approach can be used to fuse data from a video camcorder with data from set of RF sensors in the presence of occlusion.

We employ a commercial HD video camera and we downsample to standard resolution to reduce computational burden. We complement the video with a four-element multistatic RF array. Figure 1 shows the Akela radar unit and a Yagi antenna. The radar is a stepped CW type, with four ports, any of which may be used for transmit or receive. We collected data bistatically (only), yielding six unique transmit/receive antenna pairs.

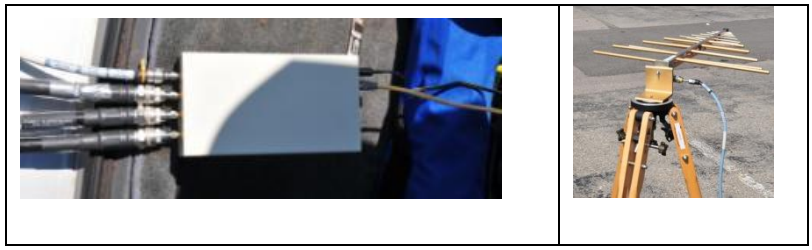

Figure 1. Experimental Radar. Left: Akela radar unit. Right: Antenna.

The camera provides information about the object's 2D position from the image pixels, and the radar provides information about the object's bistatic range and velocity from the received electromagnetic echoes.

We briefly review RF Doppler processing as a means of describing how the multistatic array provides information about target position and velocity. Figure 2 shows a cartoon depicting the radar modality, with a radar dish sending energy to, and receiving energy from, a moving object. Based on timing, individual pulses give information about range. Multiple pulses, processed over time, give information about radial velocity.

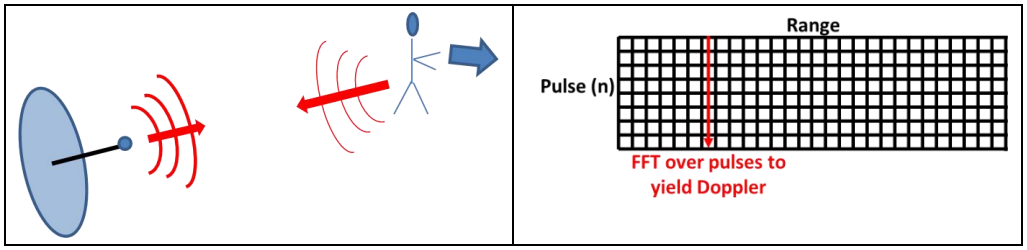

Figure 2. RF Measurements. Left: Radar dish and moving object. Right: RF data surface.

To formalize, the standard isotropic Radar scattering model describes a signal bouncing off a point object at range r0 and returning to the transmitter (monostatic operation) by

$$S = A\exp\left(-\frac{j2\pi}{\lambda}(2r_0)\right), \tag{10}$$

where A is a complex reflection amplitude, $2\pi/\lambda$ is the wavenumber for the radar centered at frequency c/λ, and the 2nd factor of 2 accounts for the RF signals' round trip propagation to (and from) an object at range r_0.

If a radar emits a sequence of N pulses, separated in time by Δt seconds (typically referred to as the Pulse Repetition Interval, or *PRI*), at an object moving with radial velocity v with respect to the radar, the signal return at pulse n is

$$S(n) = A\exp\left(-\frac{j2\pi}{\lambda}(2r_0 + nv\Delta t)\right) \tag{11}$$

Thus, the signal phase progresses in a manner proportional to the object radial velocity as $\varphi(n) = -\frac{4\pi}{\lambda}v(\Delta t)$. Radar processing capitalizes on this phenomenon by grouping *N* received pulses, Fourier transforming, and producing a Range/Doppler Map (RDM). In the experiment described below, each RDM corresponds to a discrete time step.

The maximum unambiguous phase change between consecutive pulses is $\pm\pi$. Thus, the maximum detectable radial velocity is $v_{\max} = \frac{\lambda}{4\Delta t}$. For an even number *N* of *FFT* bins, the Nyquist bin is the $(\frac{N}{2}+1)^{th}$ bin. There are $(\frac{N}{2}+1)-1=\frac{N}{2}$ intervals from the DC bin to the Nyquist bin, giving the Doppler bin resolution in units of velocity: $\Delta v_{bin} = \frac{\lambda}{4PRI} / \frac{N}{2} = \frac{\lambda}{2N\Delta t}$

In our experiments, the radar operated bistatically, meaning the transmitter and the receiver were not co-located. The relationship between Doppler bins and bistatic velocity is more complex, but follows similar principles [35].

With this as background, Figure 3 shows example input data surfaces from both the video and RF sensors. The top panel is a video frame from the camcorder. Here a moving person is evident near the bottom-right corner of the image. It is also possible to see the RF antennas in the video frame (all four are in the upper half of the frame). The bottom panel shows a range/Doppler surface (RDM) from one of the six bistatic RF antenna pairs.

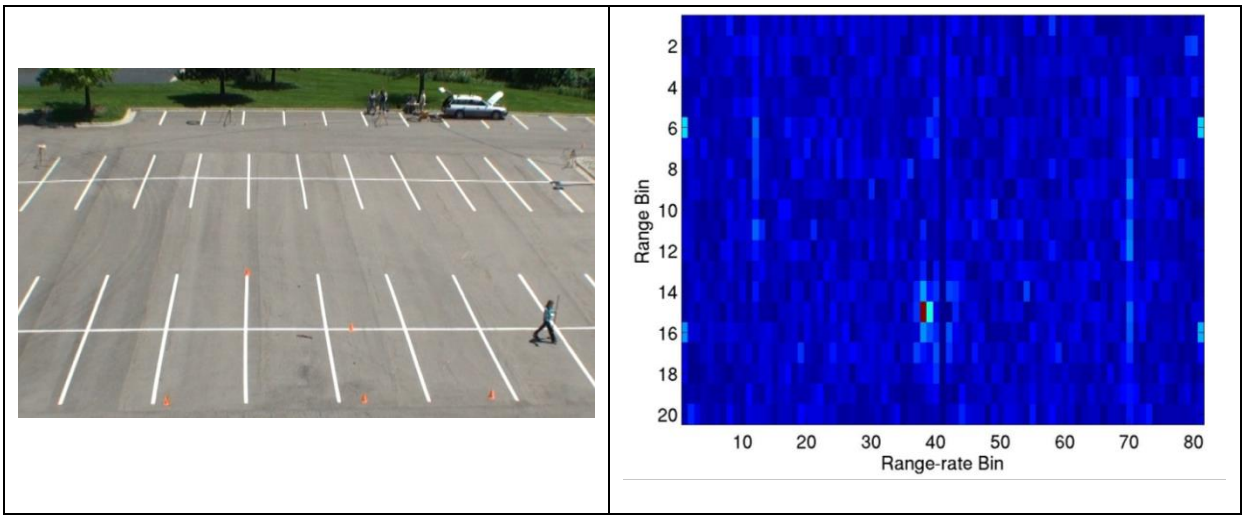

Figure 3. Data for the Experiment. Left: Single video frame. Right: RF Range/Doppler map (RDM).

We have artificially set the zero-velocity clutter bin to zero to avoid loss of dynamic range from the high clutter amplitudes. Doppler (related to range-rate, or object radial velocity) is on the horizontal axis. Range is on the vertical axis. Note that it is very difficult to gain intuition about object location on the ground or the object velocity direction because the measured quantities are bistatic with respect to a pair of antennas.

3. RESULTS

3.1 Experiment Description

In the experiment described here, a person in the parking lot of a building. Figure 4 shows a satellite image of the area around the experimental scene. Relevant to the data collected are the parking stripe patterns (also seen in the top panel of Figure 3) and the building on the left side of the image. The camcorder that collected video data stood on a tripod on the top of the building during the experiment (left side of the object scene). The RF sensors were near the right-hand end of the parking lot. Sensor areas and the object scene are marked with red text in Figure 4.

During the experiment, a person moved along a roughly triangular path in the area of the "TARGET MOTION" label from Figure 4. The four RF antennas and the camcorder simultaneously collected data during the experiment.

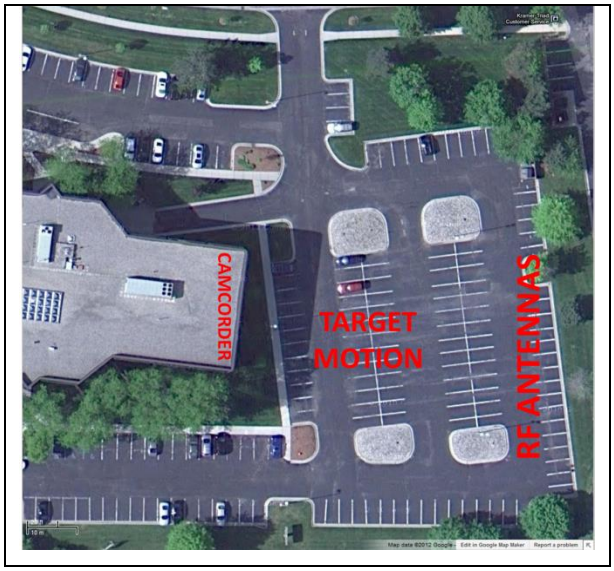

Figure 4. Satellite image of the scene

3.2 Measurement Likelihood for Video Sensor

The video data was used to compute a likelihood function. Our approach first estimated the scene background and then detected changes on a per-pixel basis over time [34]. Complex scenes with multiple objects require more sophisticated approaches to addressing dynamic object backgrounds and color features. However, in our application, we were able to implement a straightforward change detection, where we converted each video frame to grayscale and used the median pixel brightness (over time) to estimate the background image at each pixel.

Figure 5 shows the important parts of the process. The top panel shows the background ("Reference") image, accumulated across a long sequence of individual frames. The middle panel shows an example single frame. A red arrow indicates the mover. Consistent with the ideas reviewed in [34], we seek moving objects and we rely on the assumption that pixels occupied by moving objects differ significantly from (spatially) corresponding pixels in the background image. We thus create a Video Moving Target Indication (VMTI) surface. Correlation processing is a traditional approach to determining the similarity between corresponding regions in two images [36][37], and recent work has made advances in performing this computation efficiently [38]. We base our motion detection metric on this quantity, using a measure that is essentially unity minus the 2D cross-correlation between rectangular pixel regions in each frame and the each corresponding region in the reference image.

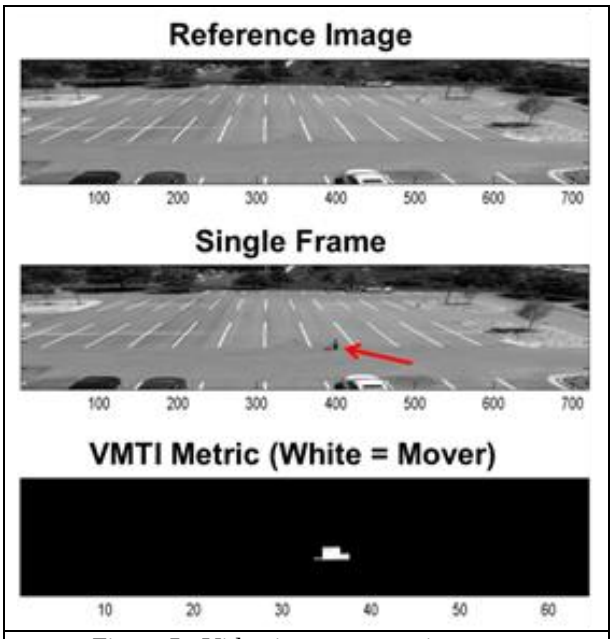

Figure 5. Video image processing steps

Figure 6 provides a cartoon example of the VMTI processing. The figure shows two images *F* and *G*. Each individual pixel (like the indicated "Pixel Under Test") defines a local neighborhood. In this example, a red box indicates the neighborhood around the test pixel. Defining these neighborhoods as f and g, respectively, the normalized 2D correlation coefficient between the neighborhoods is given by:

$$vmti(x,y) \;=\; 1 \;-\frac{1}{N}\sum_{x,y}\frac{\big(f(x,y)-\mu_f\big)\big(g(x,y)-\mu_g\big)}{\sigma_f\sigma_g} \tag{12}$$

where *x* and *y* are pixel coordinates in the neighborhood, μ_f and μ_g are the neighborhood average-values, and σ_f and σ_g are the standard deviations in the respective neighborhoods. Note that the circumstance of $f = g$ (i.e., the pixel neighborhoods is exactly the same as the reference image neighborhood), then this equation reduces to

$$\begin{aligned} vmti(x,y) \;&=\; 1 \;-\frac{1}{N}\sum_{x,y}\frac{\big(f(x,y)-\mu_f\big)^2}{\sigma_f\sigma_f} \;=\; 1 \;-\frac{1}{\sigma_f^2}\sum_{x,y}\frac{\big(f(x,y)-\mu_f\big)^2}{N} \\ &=\; 1-\frac{1}{\sigma_f^2}\sigma_f^2 \;=\; 0 \end{aligned} \tag{13}$$

Thus, for matching neighborhoods, the moving object metric is zero, as it should be.

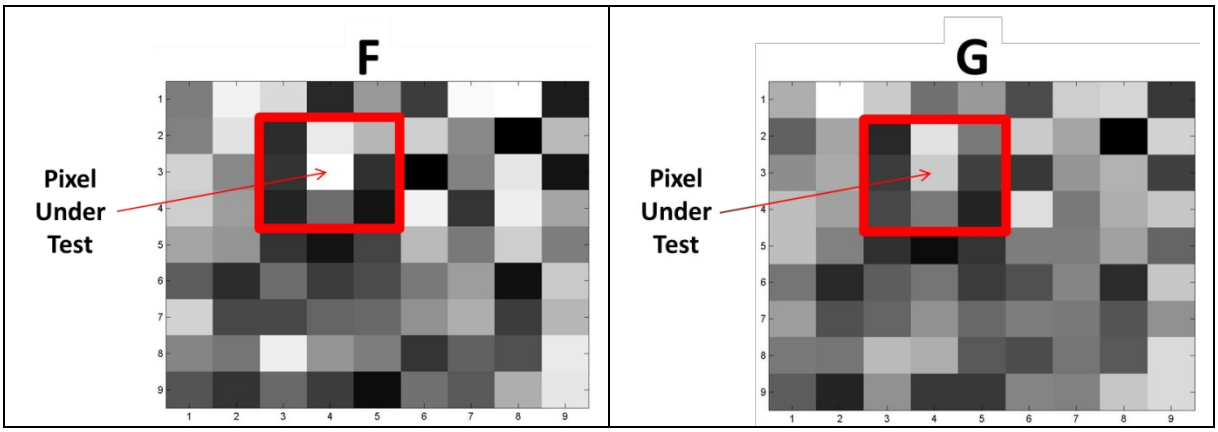

Figure 6. Cross-Correlation-based Motion Detection

The bottom panel of Figure 5 shows the VMTI metric for the frame in the middle panel. In this straightforward case, the VMTI metric is high in the region where the walking object is moving. Note (from the axis labels) that the VMTI surface comprises fewer pixels than the images. This difference stems from the fact that we down-sample as part of the correlation processing. This procedure results in a smaller, and more computationally tractable, data surface suitable for fusion and tracking.

This correlation surface is used to generate the video likelihood, described by (6), as follows. Using a calibration process, we determined target-absent and target-present pixels on the VMTI surfaces and formed empirical histograms describing the statistics when an object was present and when an object was not present.

We determined that the target-present correlation score histogram is well modeled by a Gaussian distribution. Let $j(x^k)$ denote the correlation pixel into which that hypothesized target state x^k maps. Then the notation $z_{v,j}(x^k)$ means the correlation score from the video sensor (v) in the pixel x^k maps to $j(x^k)$. The model for the pixel amplitude can then be written precisely as

$$p\left(z_{v,j(x^k)} \middle| H_1^k, x^k\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{-\left(z_{v,j(x^k)} - \mu_1\right)^2}{2\pi\sigma^2}\right) \tag{14}$$

where μ_1 is the (empirically determined) mean pixel amplitude among target-present pixels and σ^2 is the variance of these pixels.

Analyzing the target-absent amplitude histogram, we found that it closely resembles an exponential distribution, i.e.,

$$p\left(z_{v,j} \middle| H_0^k\right) = \frac{1}{\beta_0} \exp\left(-\frac{z_{v,j}}{\beta_0}\right) \tag{15}$$

where β_0 is the (empirically determined) parameter for the distribution.

Combining these two quantities yields the likelihood ratio for the video sensor

$$\begin{aligned} \Lambda_{video}\left(z_v^k \middle| x^k\right) &\propto \frac{p_{video}\left(z_{v,j(x^k)}^k \middle| H_1^k, x^k\right)}{p_{video}\left(z_{v,j(x^k)}^k \middle| H_0^k\right)} \\ &\propto \exp\left(\frac{1}{2\sigma^2}\left[z_{v,j(x^k)} - \left(\mu_1 + \frac{1}{\beta_0}\right)^2\right]\right). \end{aligned} \tag{16}$$

3.3 Measurement Likelihood for RF Array

The multistatic RF array produces a set of bistatic RDMs (range/range-rate surfaces like those shown in Figure 3) that provide information about the object's range and radial velocity. Each potential object state vector x^k couples to a unique pixel in the RDM based on its bistatic range and range-rate relative to the transmit/receive pair. We denote this pixel by $m(x^k)$. Standard radar models [35] show that the statistical distribution of energy in an RDM pixel is Rayleigh with parameter depending on target-presence vs. background (denoted as λ_t and λ_b). These parameters are estimated with a calibration process.

By analogy with the symbol definitions above, the RF likelihood can then be written

$$\begin{aligned} \Lambda_{RF}(z_{RF}^k | x^k) &\propto \frac{p\left(z_{RF,m(x^k)}^k \middle| H_1^k, x^k\right)}{p\left(z_{RF,m(x^k)}^k \middle| H_0^k\right)} \\ &\propto \exp\left(\frac{\left(z_{RF,m(x^k)}^k\right)^2 (\lambda_t^2 - \lambda_b^2)}{2\lambda_t^2 \lambda_b^2}\right) \end{aligned} \tag{17}$$

3.4 Combined Measurement Likelihood

The new data at time k is the vector measurement z^k, which includes video data z_v^k and RF data z_{RF}^k. The data updates object state estimate through the likelihood ratio $\Lambda(z^k | x^k)$. For both sensors, the algorithm computes this quantity on a per-pixel basis on the surveillance grid for use in both detection and tracking. As described in Section II(B), likelihood ratios fuse easily, according to a product rule. In all cases, $\Lambda(z^k | x^k)$ is expresses the likelihood that data z^k at state x^k came from the object-present hypothesis (object energy) rather than the object-absent hypothesis (noise and/or clutter). So as the data increasingly resembles data from an object, the likelihood ratio $\Lambda(z^k | x^k)$ increases. Figure 7 provides an example at a particular snapshot in time. Note that the figure axes correspond to target likelihoods on the ground (not in measurement space), and thus the units are in meters.

All three panels show the measurement log-likelihood ratio (simply the logarithm of the likelihood ratio), but for the different sensors. Because the RF likelihoods in this case are 4D quantities, we took marginals to eliminate the velocity (Doppler) components for display purposes. The top and middle panels, respectively, show the (spatial-only) log-likelihood ratio values for the RF sensor alone, and for the EO sensor alone. Both have an obvious region of high object likelihood. Fusing this single-snapshot data requires simply computing the product of the two individual-sensor likelihoods (as described in Section 2.2). The bottom panel demonstrates the fused likelihood for two-sensor operation. Comparing the fused likelihood with the individual sensor likelihoods shows that the fusion step reduces uncertainty about the mover's position on the surveillance grid.

In our experimental application, the RF array receives pulse data at 160 Hz. We form a coherent RDM by combining 80 pulses, yielding a measurement every 0.5s. The video sensor operates at 30 Hz. We synchronize the sources by taking every 15th frame of the video data. While this does not produce perfect time alignment, the practice is easily good enough to illustrate the data fusion processing that is the subject of this paper.

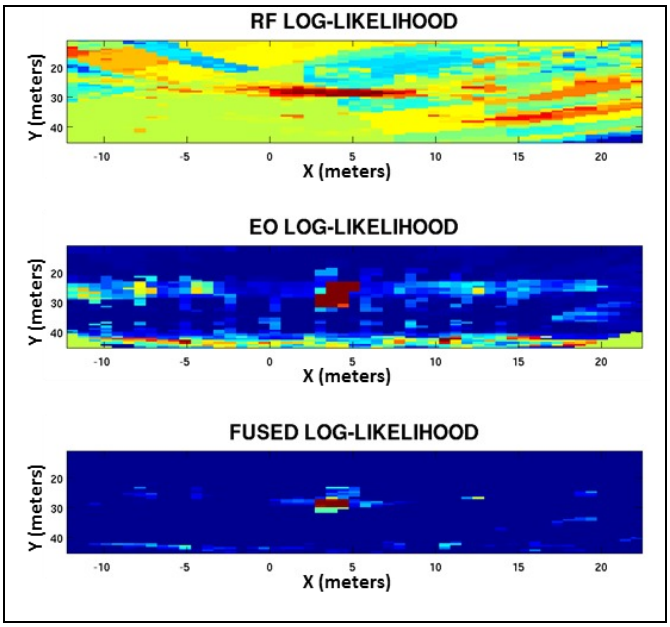

Figure 7. Individual sensor and fused log-likelihoods.

3.5 Numerical Density Estimation Approach

The sensor data does not couple linearly to the object state, meaning that the probability density of eq. (1) will not be Gaussian. Therefore, traditional tracking methods based on the Kalman Filter are not appropriate. We instead rely on a discrete approach which is a combination of fixed grid filtering and particle filtering [33]. In our discrete approach, the target state x is initially discretized onto a regular, coarse, four dimensional grid (two position states and two velocity states), which we use to coarsely estimate the density before it is known that a target is actually in the surveillance region. Next, once the coarse grid has high confidence that a target exists in the region, we transition the density estimation to a particle filter. The particle filter is able to more finely estimate the target density due to the adaptive tie points, provided that the region of interest has been sufficiently narrowed by the coarse discrete filter.

3.6 Experimental Results

Fusing data from multiple sensors offers performance improvement in at least two qualitatively different ways. Standard tracker metrics clearly demonstrate these improvements. The next two sub-sections address these two areas. First, we show that data fusion allows improved track hold in situations where one sensor is obstructed from the object. Then, we illustrate improved tracker RMSE under the circumstance that the object is visible to both sensors for the whole experiment.

Track Hold Metric

In our experiment, the mover proceeded in a triangular pattern. Both sensors collected data for the entire experiment. However, as a means of illustrating performance under obscuration, we simulated in software an obscuration blocking each sensor's view of the object at different times. These obscurations were short duration events, but their effect was to cause the trackers at each individual sensor to lose track. Figure 8 demonstrates these results.

In our experiment, we collected enough data for 62 time steps. When the obstruction occurs, there is no longer a strong source of data to pull the tracker in any direction. It therefore "coasts" with only small changes in its direction after the obscuration occurs. By the time the obscuration ends, the tracker's object position estimate is too far from the true object position for the tracker to recover. In this particular experiment, the EO obscuration occurs at time step #40 (of the 62) and persists for the duration of the experiment. In the top panel of the figure, the EO-only tracker loses track at

this point (indicated by the blue arrow) and coasts for the rest of the collection.

The RF obscuration occurs at time step #20 (indicated by the blue arrow in the lower panel). The figure shows that the RF-only tracker coasts and loses the target at the next target maneuver.

Data fusion adds significant value because the combined value of two sensors allows the tracker to hold track even when one sensor or the other is obscured. Figure 8 shows the result. Here, the RF sensor allows the tracker to hold the object while the EO sensor is obscured, and the EO sensor allows the tracker to hold the object while the RF sensor is obscured. Together, the fused sensors hold the object in track for the entire duration of the experiment.

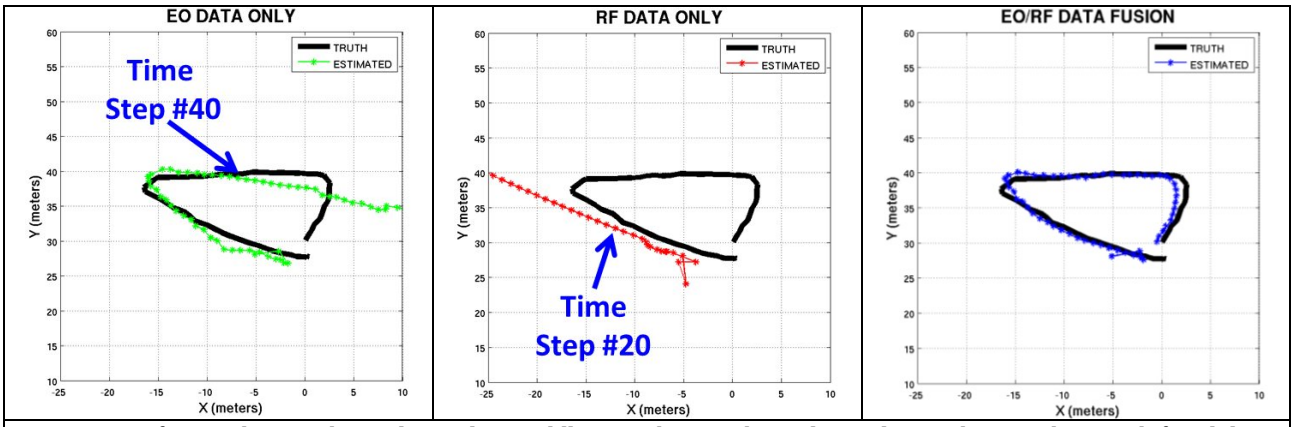

Figure 8. Left : Tracking with EO data only. Middle : Tracking with RF data only. Right : Tracking with fused data.

Tracker RMSE Metric

In the absence of occlusions, both modalities are able to sense the object and there is no track-drop issue. However, fusing data from both sensors still provides substantial benefit by reducing tracker error (quantified by RMSE). Figure 9 shows the results. In each panel, the *x*-axis corresponds to time (in discrete snapshots) and the *y*-axis shows tracker squared-error (computed at each snapshot). Each figure's title also provides the RMSE, where the mean indicates temporal averaging over snapshots.

Results are consistent with expected outcomes. The RF sensors provide the poorest spatial resolution, and thus they suffer the largest RMSE. The EO sensor (videocamera) offers fine spatial resolution and thus improved RMSE over the RF sensors alone. Despite the unequal performance of the EO and RF sensors, RF still can provide some improvement to the EO-alone situation and RMSE decreases when the processing fuses data from both modalities. In this case, improvement is quantitatively about 7% reduced error.

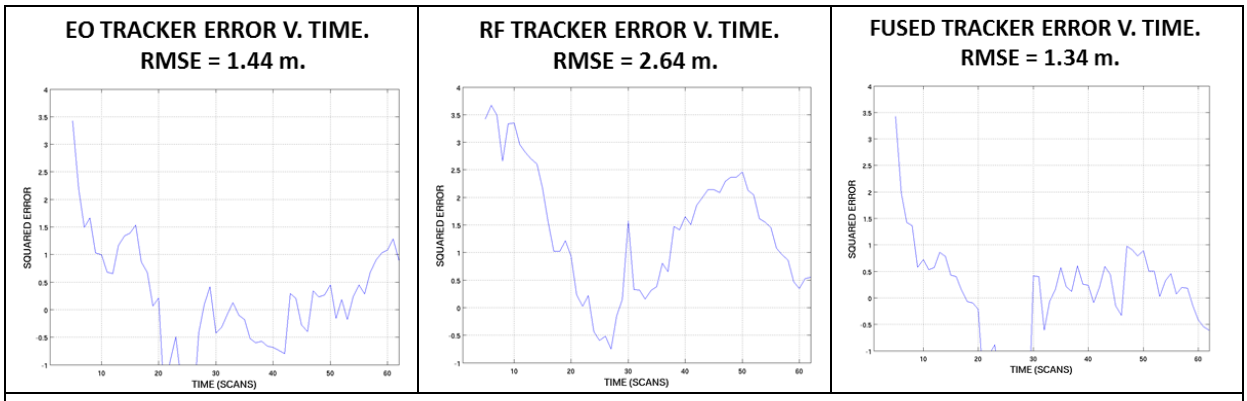

Figure 9. Tracker RMSE. Left: Video Only. Middle: RF Only. Right : Data Fusion.

4. CONCLUSION

We have presented a mathematical approach for performing data fusion within a detection and tracking system. This approach is based on Bayesian nonlinear filtering. A major advantage of the approach is that it allows fusing data from heterogeneous (including temporally asynchronous) sensors in a straightforward and natural way, at the measurement likelihood level.

We applied this fusion methodology to real, collected data that employs two modalities (video imaging and RF). Our processing ingested data from the sensors and automatically detected and tracked a moving object in a scene. We demonstrated two significant advantages offered by fusing data from the sensors prior to tracking. First, in the event that one sensor or the other might be obstructed from seeing the object, fusion allows the tracking system to maintain track-hold on the object. Second, tracker RMSE decreases when both sensors contribute to the object localization processing.

REFERENCES

[1] S.S. Blackman, and R. Popoli, Design and Analysis of Modern Tracking Systems. Artech House: Boston, MA, 1999.

[2] L.D. Stone, T.L. Corwin, and C.A. Barlow, Bayesian Multiple Target Tracking. Artech House: Boston, MA, 1999.

[3] A. Yilmaz, O. Javed, and M. Shah, M, "Object tracking: A survey," ACM Comput. Surv. 38, 4, Article 13 (Dec. 2006).

[4] Y.S. Yao, and R. Chellapa, "Tracking a Dynamic Set of Feature Points," IEEE Trans. On Image Processing, vol. 4, no. 10, pp. 1382-1395, 1995.

[5] Y. Bar-Shalom, and T. Fortmann, Tracking and Data Association. New York: Academic Press, 1988.

[6] C. Gentile, O. Camps, and M. Sznaier, "Segmentation for Robust Tracking in the Presence of Severe Occlusion," IEEE Trans. On Image Processing, vol. 13, no. 2, pp. 166-178, 2004.

[7] J. Pan, and B. Hu, "Robust Occlusion Handling in Object Tracking," 2007 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'07), pp. 1-8, Minneapolis, MN, June 2007.

[8] K. Shafique, and M. Shah, "A non-iterative greedy algorithm for multi-frame point correspondence," IEEE International Conference on Computer Vision (ICCV). 110-115, 2003.

[9] T. Darrell, G. Gordon, M. Harville, J. Woodfill, "Integrated Person Tracking Using Stereo, Color, and Pattern Detection", International Journal of Computer Vision, Volume 37, Number 2, June 2000, pp. 175-185.

[10] K. Chang, T. Zhi, R. Saha, "Performance Evaluation of Track Fusion with Information Matrix Filter," IEEE Transactions on AES, vol. 38, no. 2, April 2002.

[11] N. Joshi, S. Avidan, W. Matusik, and D.J. Kriegman, "Synthetic Aperture Tracking: Tracking through Occlusions," Proc. IEEE 11th International Conference on Computer Vision, pp. 1-8, Oct. 2007.

[12] K. Chang, T. Zhi, and R. Saha, "Performance Evaluation of Track Fusion with Information Matrix Filter," IEEE Transactions on AES, vol. 38, no. 2, April 2002.

[13] S. Coraluppi, and C. Carthel, "Recursive Track Fusion for Multisensor Surveillance," Information Fusion, vol. 5, no. 1, pp. 23–33, March 2004.

[14] S. Mori, B. Barker, C. Chong, and K. C. Chang, "Track Association and Track Fusion with Non-Deterministic Target Dynamics," Proc. IEEE Fusion 1999, pp. 231-238, July, 1999.

[15] G. Foster, "Analysis of Track Fusion using the Reduced State Estimator," Proc. IEEE Fusion 2010, July 2010.

[16] X. Tian, and Y. Bar-Shalom, "Exact algorithms for four track-to-track fusion configurations: All you wanted to know but were afraid to ask," Proc. IEEE Fusion 2009, pp. 537-544, July, 2009.

[17] R. Canavan, C. McCullough, and W. Farrell, "Track-centric metrics for track fusion systems," Proc. IEEE Fusion 2009, pp. 1147-1154, July, 2009.

[18] C. Yang, and E. Blasch, "Track Fusion with Road Constraints," Proc. IEEE Fusion 2007, July, 2007.

[19] H. Chen, and X. R. Li, "On track fusion with communication constraints," Proc. IEEE Fusion 2007, July 2007.

[20] Ben Shapo, and Chris Kreucher, "Track-Before-Fuse Error Bounds for Tracking Passive Targets," 2011 Proceedings of the International Conference on Information Fusion, IEEE Catalog Number CFP11FUS-PRT, July 2011.

[21] R. Bethel, and G. Paras, "A PDF multisensor Multitarget Tracker," IEEE Transactions on AES, vol. 34, no. 1, pp. 153–168, January 1998.
[22] C. A. Barlow, L. D. Stone, and M. V. Finn, "Unified Data Fusion," in Proceedings of the Ninth National Symposium on Sensor Fusion, March 1996.
[23] K. Kastella, "Joint Multitarget Probabilities for Detection and Tracking," in Proceedings of SPIE Acquisition, Tracking and Pointing XI, 1997.
[24] A. Srivastava, M. Miller, and U. Grenander, "Jump-diffusion Processes for Object Tracking and Direction Finding," Proc. of 29th Allerton Conf. on Communication, Control, and Computing, 1991, pp. 563 – 570.
[25] E. Kamen, "Multiple Target Tracking Based on Symmetric Measurement Functions," IEEE Transactions on Automatic Control, vol. 37, no. 3, pp. 371–374, 1992.
[26] S. Dockstader, and M. Tekalp, "Multiple Camera Tracking of Interacting and Occluded Human Motion," Proceedings of the IEEE, vol. 89, no. 10, October 2001.
[27] J. Kang, I. Cohen and G. Medioni. "Multi-Views Tracking Within and Across Uncalibrated Camera Streams," ACM SIGMM Workshop on Video Surveillance, 2003.
[28] K. Kim, and L.S. Davis, "Multi-Camera Tracking and Segmentation of Occluded People on Ground Plane using Search-Guided Particle Filtering," European Conference on Computer Vision (ECCV), LNCS, 2006.
[29] A. O. Ercan, A. El Gamal and L. J. Guibas, "Object Tracking in the Presence of Occlusions via a Camera Network," Proceedings of IPSN, Cambridge, MA, April 2007.
[30] D.P. Williams, "Bayesian Data Fusion of Multiview Synthetic Aperture Sonar Imagery for Seabed Classification," IEEE. Trans. On Image Processing, vol. 18, no. 6, June 2009.
[31] J. Kang, K. Gajera, I. Cohen and G. Medioni. "Detection and Tracking of Moving Objects from Overlapping EO and IR Sensors," Joint IEEE International Workshop on Object Tracking and Classification Beyond the Visible Spectrum (OTCBVS'04). Washington, D.C. June, 2004.
[32] B. Shapo, and R. E. Bethel, "An Overview of the Probability Density Function (PDF) Tracker," Proceedings of the Oceans 2006 Conference, Boston, Sept. 2006.
[33] C. Kreucher, and B. Shapo, "Multitarget Detection and Tracking using Multi-Sensor Passive Acoustic Data", IEEE Journal of Oceanic Engineering, 36(2):205-218, April 2011.
[34] Y. Sheikh, and M. Shah, "Bayesian modeling of dynamic scenes for object detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(11): 1778–92.
[35] C. Kreucher, "Exploiting Narrowband Bistatic Radar Measurements for Dismount Detection and Tracking". IEEE Antennas and Propagation Magazine, 53(1):98-105, February 2011.
[36] Brown, L. G., "A survey of image registration techniques," ACM Comput. Surv. 24(4), 325–376 (1992).
[37] Zitova, B., and Flusser, J., "Image registration methods: a survey," Image and Vision Computing 21, 977–1000 (2003).
[38] X. Sun, N. P. Pitsianis, and P. Bientinesi, "Fast computation of local correlation coefficients," Proc. SPIE 7074, 707405 (2008).
[39] Kyle M. Tarplee, David J. Trawick, and Shawn M. Herman, "Distributed multiple-hypothesis correlation and feedback with applications to video data", Proceedings of SPIE, vol. 6969, 2008.
[40] Ranga Narayanaswami, Anastasia Tyurina, David Diel, Raman K. Mehra, and Janice M. Chinn, "Discrimination and tracking of dismounts using low-resolution aerial video sequences", Procedings of SPIE, vol. 8137, 2011.