

# Feature-Aided Multiple Target Tracking in the Image Plane

Andrew P. Brown<sup>\*a</sup>, Kevin J. Sullivan<sup>a</sup>, David J. Miller<sup>b</sup>

<sup>a</sup>Toyon Research Corporation, 6800 Cortona Drive, Goleta, CA 93117;

<sup>b</sup>Pennsylvania State University, 227C EE West, University Park, PA 16802

## ABSTRACT

Vast quantities of EO and IR data are collected on airborne platforms (manned and unmanned) and terrestrial platforms (including fixed installations, e.g., at street intersections), and can be exploited to aid in the global war on terrorism. However, intelligent preprocessing is required to enable operator efficiency and to provide commanders with actionable target information. To this end, we have developed an image plane tracker which automatically detects and tracks multiple targets in image sequences using both motion and feature information. The effects of platform and camera motion are compensated via image registration, and a novel change detection algorithm is applied for accurate moving target detection. The contiguous pixel blob on each moving target is segmented for use in target feature extraction and model learning. Feature-based target location measurements are used for tracking through move-stop-move maneuvers, close target spacing, and occlusion. Effective clutter suppression is achieved using joint probabilistic data association (JPDA), and confirmed target tracks are indicated for further processing or operator review. In this paper we describe the algorithms implemented in the image plane tracker and present performance results obtained with video clips from the DARPA VIVID program data collection and from a miniature unmanned aerial vehicle (UAV) flight.

**Keywords:** Feature-aided multiple target image/video tracking

## 1. INTRODUCTION

The image plane tracker (IPT) processes input video frames to generate *tracks*—sequences of position and velocity estimates—for potential targets of interest (objects such as vehicles or people). The target positions are specified in the 2D image plane, with the continuous-valued doublet  $(x \text{ [column]}, y \text{ [row]})$ , in pixels. The coordinate system is defined as right-handed, with the origin at the image center<sup>\*\*</sup>. The choice of tracking in the image plane is motivated by the need for computational efficiency in processing the huge amount of data provided by typical video sensors. If the platform location and orientation—and the camera parameters—are known, then it is straightforward to calculate the target locations in 3D world coordinates, as required for multi-platform data fusion and for many applications, including aerial surveillance, reconnaissance, and tactical fire control. By processing high-frame-rate, high-resolution video in the image plane, a wealth of information including target location, movement, and appearance—simultaneously for a large number of targets—is provided to assist operators and commanders in most effectively utilizing the video platform assets and achieving their mission objectives.

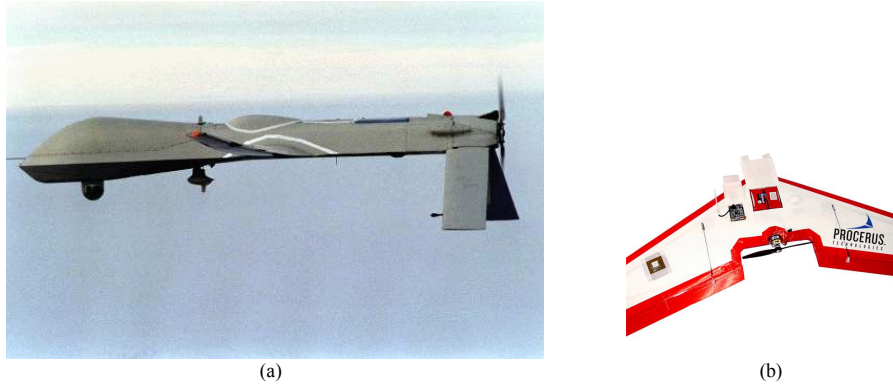
One of many promising applications for the image plane tracker is video processing in airborne surveillance. With an IPT running locally on each platform and automatically detecting and tracking targets on the ground, pilot/operator efficiency and effectiveness could be greatly enhanced. For UAV platforms, a higher level of autonomy could be achieved, with essential target information provided directly to the platform controller for use in route adjustment and collaborative platform/sensor tasking decisions. This level of autonomy could allow operators to supervise multiple UAVs based on conveniently displayed track information. The UAV platforms could be large, high-altitude assets such as the Predator (Fig. 1(a)), or miniature UAVs such as the 24-oz., 48-in. wingspan Zagi<sup>1,2</sup> (Fig. 1(b)). In this paper, the performance of the IPT is demonstrated using results from two representative video clips: EgTest02 (collected on a stable, high-altitude platform) from the VIVID Tracking Evaluation Web Site<sup>3</sup>, and from the Procerus<sup>TM</sup> “Convoy Following” Zagi flight<sup>2</sup>. The IPT tracking results are excellent for both videos, with track confirmation probability of

---

\* abrown@toyon.com; phone 1 805 968-6787x101; fax 1 805 685-8089; toyon.com

\*\* In the right-handed coordinate system, the row index ascends from the lower to the upper part of the image. Positive rotation is counter-clockwise about the center of the image.

1.0 for objects moving at speeds greater than 2 m/sec., no confirmed false tracks, and 100% track continuity. Fig. 2 provides screenshots with example tracking performance for Convoy Following. Confirmed tracks are indicated by yellow bounding boxes surrounding the targets, with track label\* superscripts.



**Figure 1:** Example platforms for the image plane tracker: the Predator (a) and Zagi<sup>1,2</sup> (b) UAVs.



**Figure 2:** Screenshots from the image plane tracker display for the Procerus™ Convoy Following Zagi flight.

The following sections describe the primary functions performed by the image plane tracker: (a) frame registration/image prediction and track prediction, (b) change detection for moving target detection and segmentation, (c) JPDA track updating and track database maintenance, and (d) utilization of target feature information. Briefly, frame registration compensates for platform and camera motion by aligning consecutive video frames such that pixels with the

\* When targets exit and reenter the field of view, they may be assigned different labels, since the IPT only tracks targets within the field of view. Fusion of track information across multiple fields of view can be performed by one or more fusion nodes to which the local IPTs send track and target feature information.

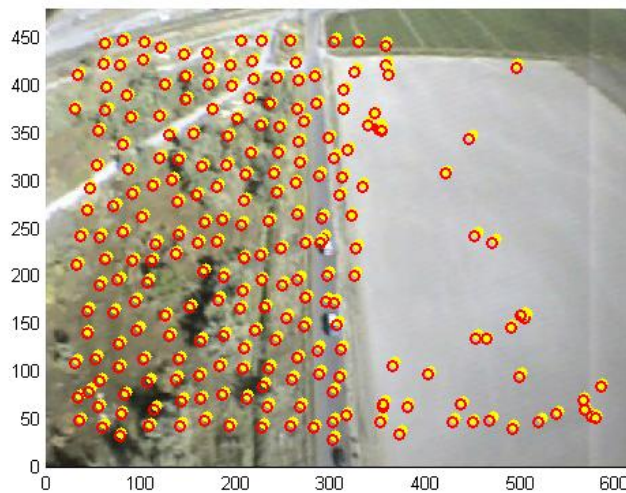
same indices correspond to the same points in the scene, as required for change detection. The registration parameters are also used to compensate for platform motion when predicting tracks into the next frame, using the Kalman filter. Change detection based on computationally efficient background modeling and difference image filtering is used to classify each image pixel as stationary background or moving foreground, as required for motion-based target position measurement and target blob segmentation. Information from segmented target blobs is used to learn target appearance features, which are subsequently used to obtain target location measurements for both moving and stationary targets. The target feature information extracted by the IPT can also be used in track-to-track fusion across multiple platforms and for maintaining track continuity in the presence of closely spaced targets and occlusion. In the IPT, both motion- and feature-based target location measurements are combined in the JPDA update step for joint multiple target tracking and clutter rejection. Track maintenance tasks include candidate track initialization, lost/false track termination, good track confirmation, and visual display of tracking information.

## 2. FRAME REGISTRATION AND TRACK PREDICTION

To enable high-fidelity change detection and track prediction, video collected with a moving platform/camera must be stabilized, i.e., consecutive frames must be aligned/registered such that pixels with the same indices correspond to the same points in the scene. Under the far-field assumption, the effect on the image background (excluding any moving foreground targets) can be represented via an affine transformation, which is able to account for the effects of translation, rotation, and non-isotropic scaling. In this model, the location of an image point is transformed from frame  $t-1$  to  $t$  as

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \mathbf{D} \begin{bmatrix} x(t-1) \\ y(t-1) \end{bmatrix} + \mathbf{d}, \text{ with parameters } \mathbf{D} = \begin{bmatrix} d_{xx} & d_{xy} \\ d_{yx} & d_{yy} \end{bmatrix}, \mathbf{d} = \begin{bmatrix} d_x \\ d_y \end{bmatrix}. \quad (1)$$

Estimates of platform and camera motion will often be available from inertial and GPS measurements, as well as from known control inputs. However, to achieve the desired frame registration accuracy, it is necessary in many applications to utilize information from the measured image sequence in estimating the affine transformation parameters. The IPT computes a least-squares estimate of the affine transformation parameters, based on the observed motion of background features. To obtain the background feature locations and correspondences between frames, the IPT employs the Kanade-Lucas-Tomasi (KLT) Feature Tracker<sup>4,5,6</sup> open source software<sup>7</sup>, as well as some supporting functions provided in the CTracker<sup>8</sup> open source software. In the first frame, KLT searches for background corner features (i.e., small image regions which satisfy a constraint on the filtered intensity gradient). The located corner features are efficiently tracked from frame to frame using Newton-Raphson iteration. Features are eliminated if they are judged—based on feature match residuals—to have been poorly tracked. CTracker then uses an outlier rejection algorithm based on Random



**Figure 3:** Corner feature points in the previous (closed yellow circles) and current (open red circles) frames that agree on a common displacement are superimposed on the current frame.

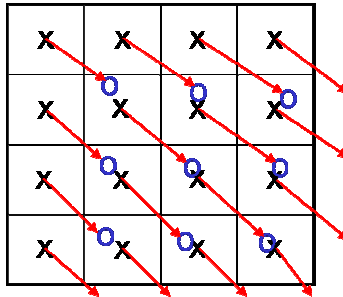
Sample Consensus (RANSAC)<sup>9</sup> to find feature correspondences that agree on a common translation. To counteract feature depletion, new features are added as needed. An example of the resulting feature correspondences is provided in Fig. 3, with features in frame  $t-1$  and  $t$  denoted by yellow closed circles and red open circles, respectively.

The image plane tracker estimates the affine transformation parameters using the measured corner feature locations  $(x_i(t-1), y_i(t-1))$  and  $(x_i(t), y_i(t))$  in the previous and current frames, respectively, for feature points  $i = 1, 2, \dots$ . The system of linear equations solved via least-squares is

$$\begin{bmatrix} x_1(t-1) & y_1(t-1) & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1(t-1) & y_1(t-1) & 1 \\ x_2(t-1) & y_2(t-1) & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2(t-1) & y_2(t-1) & 1 \\ x_3(t-1) & y_3(t-1) & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_3(t-1) & y_3(t-1) & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} * \begin{bmatrix} d_{xx} \\ d_{xy} \\ d_x \\ d_{yx} \\ d_{yy} \\ d_y \end{bmatrix} = \begin{bmatrix} x_1(t) \\ y_1(t) \\ x_2(t) \\ y_2(t) \\ x_3(t) \\ y_3(t) \\ \vdots \end{bmatrix}. \quad (2)$$

## 2.1 Frame registration

Alignment/registration of the images requires 2D image interpolation in each of the spectral bands (e.g., red, green, and blue, or IR). This is because pixels located at discrete locations (row, column indices) in one image are in general located at continuous locations in the next image, as illustrated in Fig. 4. For computational efficiency, the IPT processing is arranged such that for alignment of frames  $t-L$  and  $t$ , for  $L \geq 1$ ,  $(x(t-L), y(t-L))$  are continuous-valued and  $(x(t), y(t))$  are discrete-valued.

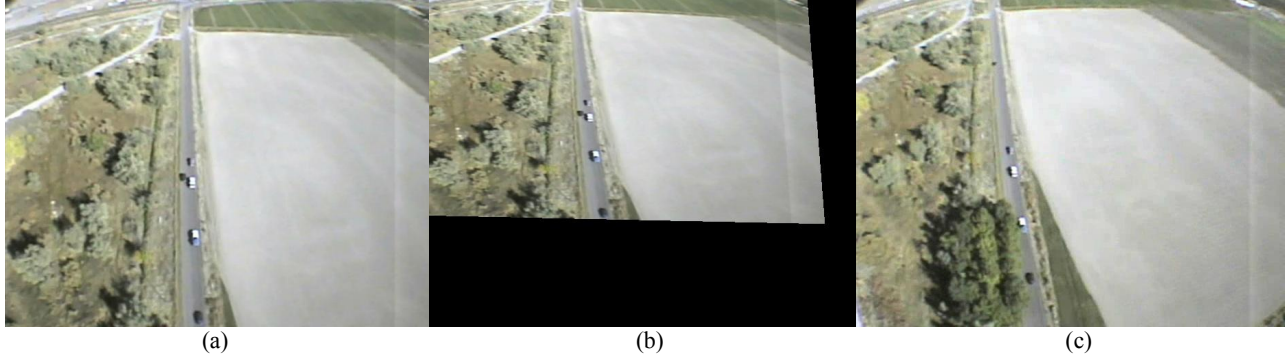


**Figure 4:** Pixels located at discrete indices (X's) in one image are located at continuous locations (O's) in the next frame. Thus, image registration requires 2D image interpolation.

To demonstrate the accuracy with which frame alignment was performed for the miniature UAV platform (which like any very light weight aircraft is rather unstable), cumulative frame alignment was performed over a lag of  $L = 100$  frames, with the affine transformation parameters computed recursively from the 1-lag values. The inverse alignment of frame 101 with frame 1 is shown in Fig. 5. Approximately 95%, on average, of the apparent image motion—including scale change—is accounted for.

## 2.2 Track prediction

In image plane tracking, track prediction must account for apparent target motion due to platform/camera motion, as well as for actual target motion. For the target state dynamics model, the IPT assumes the discrete white noise acceleration model<sup>10</sup>. From (1), the combined effect on the target state *in the image plane*, can be modeled as



**Figure 5:** To demonstrate frame alignment accuracy, frame 101 (a) is aligned via the inverse affine transform (b) with frame 1 (c).

$$\begin{bmatrix} x(t) \\ y(t) \\ v_x(t) \\ v_y(t) \end{bmatrix} \cong \mathbf{F}(t, t-1) \begin{bmatrix} \hat{\mathbf{D}}(t, t-1) & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{D}}(t, t-1) \end{bmatrix} \begin{bmatrix} x(t-1) \\ y(t-1) \\ v_x(t-1) \\ v_y(t-1) \end{bmatrix} + \mathbf{F}(t, t-1) \begin{bmatrix} \hat{\mathbf{d}}(t, t-1) \\ \mathbf{0} \end{bmatrix} + \mathbf{w}(t-1), \quad (3)$$

where  $\mathbf{F}(t, t-1)$  is the state transition matrix,  $\mathbf{w}(t-1)$  is the process noise, and  $\hat{\mathbf{D}}(t, t-1)$ ,  $\hat{\mathbf{d}}(t, t-1)$  are the affine transformation parameter estimates. Given (3), track prediction is performed using the standard Kalman filter prediction equations.

### 3. MOVING TARGET DETECTION AND SEGMENTATION

Moving target detection and segmentation can be achieved using change detection, i.e., by comparing the current frame with the previous frame—or multiple frames—to locate regions where change is occurring due to foreground target motion. A comprehensive survey of the state-of-the-art is given in<sup>11</sup>. A popular approach<sup>12,13</sup> to change detection is to model the variation of each pixel as a random pixel process, with a probability distribution approximated by a mixture of Gaussian densities representing both background and foreground pixel states. However, the computational complexity of this approach is prohibitive in many applications. Furthermore, it may be infeasible to build detailed multi-modal statistical models of scene appearances, given the high speeds at which many airborne platforms travel, and hence the fast-changing field of view. Most critically, these approaches to scene background modeling are not ideally suited to mitigate the two most prominent difficulties with change detection from an airborne platform: (1) apparent change due to frame registration errors, and (2) apparent change due to parallax, whereby the appearance of extended objects changes as they are viewed from different angles. In contrast, our image plane tracker employs a novel, computationally efficient algorithm for adaptively estimating the typical apparent *change* in each background region, which at the pixel level is a more stable quantity than the *appearance* of the region. Given the typical apparent change model (TACM) for the scene background—which excludes the effects of foreground target motion—standard frame difference measurements are found to be of great utility for accurate, efficient change detection.

Frame difference measurements are obtained as follows. In each spectral band  $j = 1, \dots, N_j$ , the difference at each  $(x(t), y(t))$  is computed:

$$d_j(x(t), y(t)) = I_j^c(x(t), y(t)) - I_j^p(x(t-L), y(t-L)), \quad (4)$$

where  $I_j^c(x(t), y(t))$  and  $I_j^p(x(t-L), y(t-L))$  are the current and previous (registered) image intensities in the  $j$ -th spectral band. Prior to differencing, the images may be normalized to have the same mean and variance. Typically,  $L = 1$  (consecutive images are differenced), but larger values of  $L$  can be used for detection of slow-moving targets.



Under the assumption that *in the image background* the  $d_j(x(t), y(t))$  are distributed zero-mean Gaussian with variance  $\sigma_j^2(t)$ , the normalized difference measurement

$$z(x(t), y(t)) = \sum_{j=1}^{N_j} d_j(x(t), y(t))^2 / \sigma_j^2(t) \quad (5)$$

is central chi-square distributed with  $N_j$  degrees of freedom. A reasonable estimate of  $\sigma_j^2(t)$  is given by

$$\hat{\sigma}_j^2(t) = \operatorname{median}_{x,y} \left\{ \frac{1}{N_j} \sum_{j=1}^{N_j} d_j(x(t), y(t))^2 \right\}, \quad (6)$$

where the median operator performs outlier rejection on contributions from moving foreground objects (the ratio of the number of background to foreground pixels is assumed  $> 1$ ). Estimation of a new normalization factor  $\sigma_j^2(t)$  in each frame provides robustness to variations in the amount of apparent background change due to uncompensated platform/camera motion and changes in scene illumination, etc. With this motivation, a reasonable decision rule for each pixel in the difference image  $z(x(t), y(t))$  is given by:

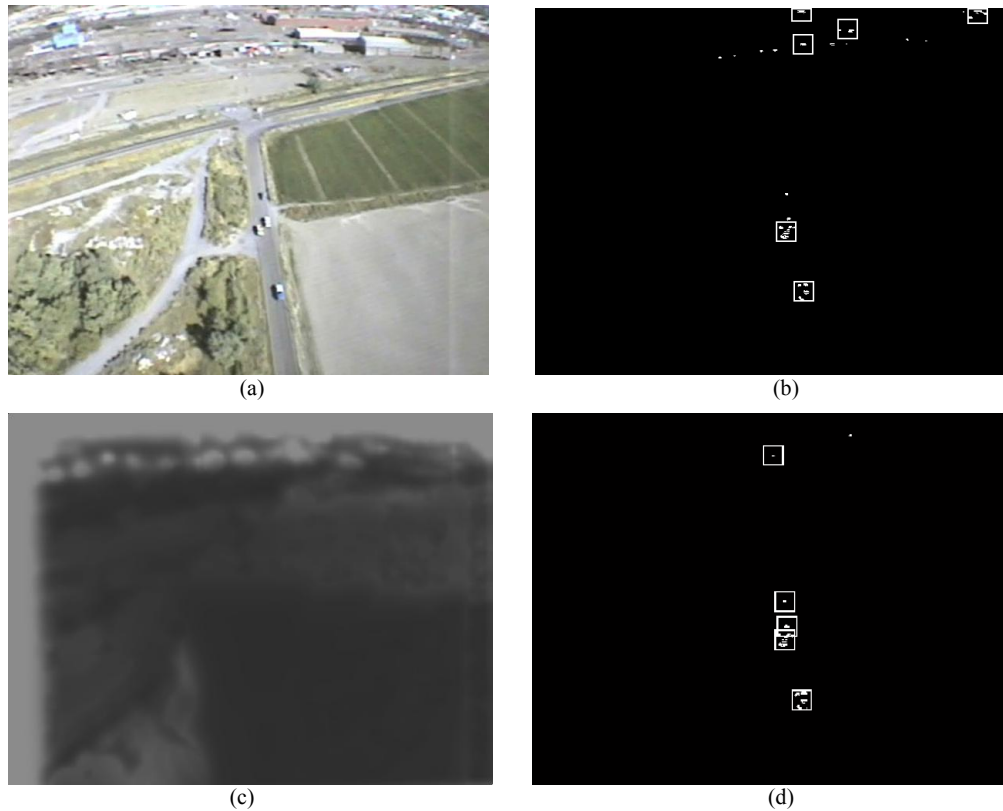
$$z(x(t), y(t)) > / < \lambda_z \rightarrow \text{"foreground"} / \text{"background"}, \quad (7)$$

where the threshold  $\lambda_z$  is set by consulting tables on the chi-square cumulative distribution function. In practice, the Gaussian assumption on  $d_j(x(t), y(t))$  is weak, since the distribution tends to be heavy-tailed. This necessitates the choice of larger values of  $\lambda_z$ . A representative result for the change detection mask, using the simple classification rule (7), is shown in Fig. 6(b), with white pixels corresponding to foreground/motion classifications and black pixels corresponding to stationary/background classifications. The white bounding boxes indicate moving target detections, as discussed below. Comparing the change detection mask with the raw frame in Fig. 6(a), only two out of four moving vehicles were detected, with four false alarm detections. This typical performance illustrates the problem with simple frame differencing, namely that imperfect frame registration on highly-mobile platforms can result in large apparent change in small image regions, compared with change due to object motion.

Our solution to the problem of localized apparent change is to adaptively compute the typical apparent change model (TACM), which is a statistical model of the apparent *background* change in local image regions, and is ideally independent of foreground motion. TACM is implemented in the image plane tracker with a truly negligible increase in computation and storage requirements. Fig. 6(c) shows the TACM model corresponding to Fig. 6(a). Lighter regions in the image correspond to regions with greater typical apparent change in the background scene, while darker regions correspond to lesser typical apparent change. Note that the effect of the moving foreground vehicles is effectively excluded from the TACM. With difference image normalization based on the TACM, rather than the median background normalizer, the decision rule (7) yields much-improved change detection performance, as shown in Fig. 6(d). In this representative result, all four moving vehicles are detected, with only one false alarm detection. Because of the effectiveness of TACM, the image plane tracker is able to achieve excellent tracking performance, even in the presence of significant camera aimpoint instability.

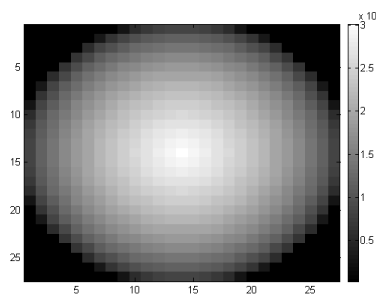
### 3.1 Moving target position measurement

Given the change mask, moving foreground objects can be detected, and their positions measured. While the change mask provides individual pixel classifications, the detection function performs higher-level processing to identify the cluster of pixels corresponding to each moving foreground object. The first step is to remove speckle noise, corresponding to the apparent motion of objects which are too small to be candidate targets of interest. This is



**Figure 6:** For the current image frame (a), the change detection masks and target detection bounding boxes are shown with (d) and without (b) the use of an adaptive typical apparent change model (TACM).

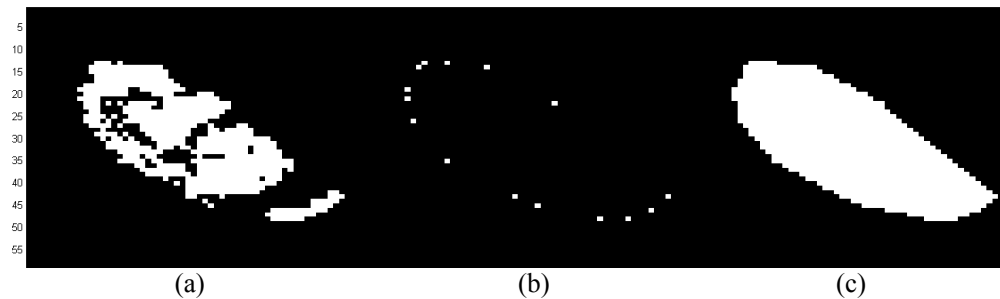
accomplished via multiplication with a clutter rejection mask obtained as the thresholded and saturated convolution of the change mask with a small  $N$ -by- $N$  (e.g.,  $N = 5$ ) kernel of “1”s. The cluster of pixels corresponding to each moving foreground object is then identified by convolving the change mask with the clustering kernel shown in Fig. 7. The kernel’s diameter is equal to the target bounding box size, set to the maximum expected moving object dimension (which is estimated adaptively across the frame sequence). Because the kernel gives more weight to change mask pixels closer to the kernel’s center, peaks in the convolved change image correspond to the likely locations of moving object centers of mass. The location of the highest peak is selected as a target detection and position measurement, and the change mask pixels within the bounding box are zeroed out. Subsequent target detections and location measurements are performed for the remaining highest peaks until nearly all (95%) of the change mask foreground pixels are accounted for.



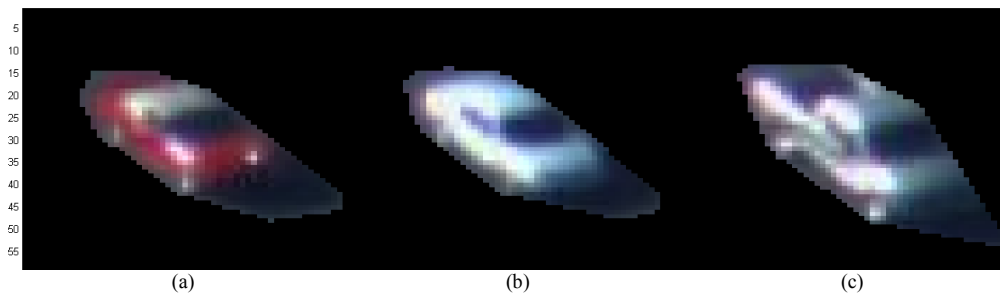
**Fig. 7:** The clustering kernel, with diameter equal to the maximum expected moving object dimension, is weighted proportionally to the inverse of the squared distance from the kernel center.

### 3.2 Moving target segmentation

For each detected moving target, the pixels on the target—collectively, the target “blob”—are segmented for use in target feature modeling, and feature-based measurement of moving and stationary target locations, as discussed in the following section. In determining which pixels correspond to the contiguous target blob, it is insufficient to simply consider foreground pixel connectivity. This is because interior target regions are often not classified as foreground, due to similar coloring across the target surfaces. As shown in Fig. 8, within the bounding box the change mask foreground pixels (a) are processed to find those pixels lying on the convex hull (b), and all pixels within the convex hull are taken to be the segmented target blob pixel mask (c). Example segmented target blobs are shown in Figure 9. These results are for the EgTest02 video, which is used for the remainder of the results presented in this paper. Two tracked screen shots from this video are shown in Figs. 10(a),(b), illustrating the significant change in scale between the beginning and end of the video (43 sec. duration). The change in scale is tracked by adaptively estimating the target bounding box size based on the areas of the segmented target blobs.



**Figure 8:** Moving target segmentation: (a) change mask foreground pixels, (b) convex hull, and (c) target blob pixel mask.



**Figure 9:** Example segmented blobs for a red Ford Mustang (a), silver Chevy Cavalier (b), and tan Dodge Ram SLT (c).



**Figure 10:** Substantial change in scale occurs during the 43 sec.-duration EgTest02 video. The change in scale is tracked by estimating the target bounding box sizes from the areas of the segmented target blobs.



#### 4. TRACK UPDATING AND DATABASE MAINTENANCE

Each track state consists of position and velocity components in  $x$ - $y$  image coordinates. Thus, one method of track initialization is to use position measurements from two consecutive frames to initialize the track position and velocity estimates. However, for high-clutter environments, we find it convenient and effective to initialize tracks using the measurement locations in the first frame, with zero initial velocity estimates, and large standard deviations. Measurement-to-track associations in the second and following frames are computed probabilistically using JPDA<sup>14</sup>, which performs each track update using an association probability-weighted combination of the measurements and the missed detection hypothesis. Validation gating is used to limit the number of tracks with which a measurement is associated. Measurements are also probabilistically associated with the “clutter” hypothesis. In the IPT, measurements that are associated in high probability with the clutter hypothesis are used to initialize new tracks, since these measurements may originate from true moving targets which are not yet in track.

When the IPT gains enough confidence in its location estimate for a target in track (that is, when the maximal extent of the track error ellipse—as calculated from the track covariance—becomes small enough), the track is confirmed, and the track bounding box is displayed, enclosing the target. Conversely, if the confidence in a location estimate becomes sufficiently poor, the track is judged to be false (i.e., to correspond to clutter, rather than to a true target), or to be lost (e.g., due to extended occlusion). The IPT also chooses to terminate tracks when the corresponding targets leave the field of view, and to initialize new tracks if the targets reenter the field of view. This is illustrated in Fig. 11, where in the brief interval between the frames shown in (a) and (b), the camera rapidly pans to the right and then back to the left, causing the three targets on the left to exit and then reenter the field of view. In (b), the IPT has begun the process of re-confirming the tracks, which typically takes on the order of 1 sec. Even though the camera quickly pans from side to side several times, the tracks labeled “1,” “2,” and “3” are never shaken off the corresponding targets, which remain in the field of view. The robustness of the IPT to rapid camera panning provides evidence of the effectiveness of the frame registration algorithms.



**Figure 11:** Rapid camera panning between frames (a) and (b) causes three targets to exit and then reenter the field of view. Tracks are dropped when targets exit the field of view and are reinitialized when targets reenter. Rapid camera panning is effectively accounted for via frame registration.

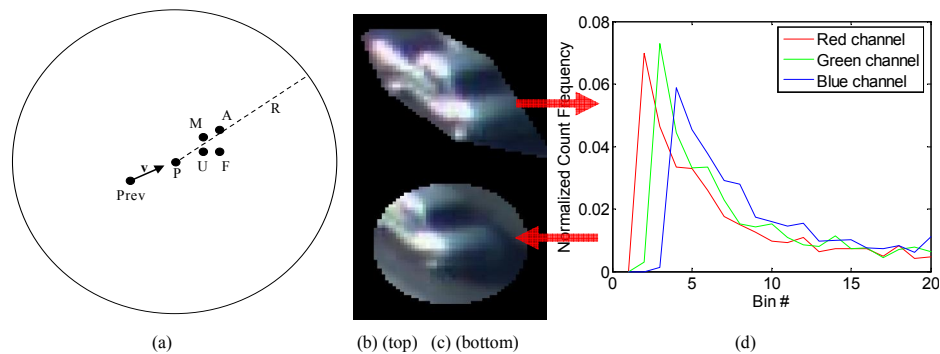
#### 5. TARGET FEATURE INFORMATION

In the image plane tracker, information from segmented target blobs is used to learn target appearance feature models. In subsequent frames, the feature models are then used to obtain target location measurements for both moving and stationary targets. The availability of these feature-based measurements has proven to be very valuable for tracking multiple closely-spaced targets through move-stop-move conditions. The target feature information extracted by the IPT can also be used in feature-aided tracking<sup>15,16</sup> for improved track continuity in the presence of closely spaced targets and occlusion, as well as for track-to-track fusion across multiple platforms.

In the IPT, the currently-implemented feature choice is the spectral (e.g., RGB or IR intensity) histogram (Fig. 12(d)) computed from the segmented target image blob (Fig. 12(b)), assuming the blob is JPDA-associated with high probability with a single existing track. The histogram model is updated as a forgetting-factor-based average over the current and past feature observations. The choice of spectral histogram is motivated by its quasi-invariance to scale, target pose, and viewing angle changes.

Given the track-specific feature model from the previous frame, the target coordinates in the current frame are identified by searching over a circular region of radius  $R$  (e.g., half the bounding box size) centered on the predicted target coordinates,  $P$ . As illustrated in Fig. 12(a),  $P$  is obtained from the previous frame location estimate,  $Prev$ , using the estimated target velocity,  $v$ , and the estimated affine transformation parameters from frame registration. The motion- and feature-based target coordinate measurements,  $M$  and  $F$ , respectively, lie near the actual target coordinates,  $A$ . Using measurements  $M$  and  $F$ , the Kalman filter-updated target coordinates are  $U$ .

The feature-based measurement  $F$  is obtained as follows. Candidate target centroid coordinates are generated as coarsely-spaced integer pixel shifts in the region shown in Fig. 12(a). For each candidate centroid, the spectral histogram is computed in a circular segmentation region surrounding the centroid, as shown in Fig. 12(c). The radius of the circular segmentation region is track-specific and is learned from the data (the areas of the segmented blobs from previous frames). Note that using circular segmentation provides semi-invariance to changes in target pose and viewing angle. The candidate centroid for which the observed histogram lies closest (in terms of minimum Euclidian distance in the feature space) to the model histogram (and subject to a threshold on match quality) is selected as  $F$ . While the motion-based measurements ( $M$ s) are associated *in probability* with each track in the JPDA framework, the feature-based measurement  $F$  is hard-associated with a track, since the track kinematics are used to define the limited search region, and because the feature match is computed using the track-specific feature model.



**Figure 12:** Spectral histogram model learning and feature-based location measurement for moving and stationary targets: In (a), a sparse search for the best feature match is performed within a radius  $R$  of the predicted target location, in (b) the segmented blob from a moving target is used to update the track-specific feature model (d), which is matched with candidate target blobs (c) in subsequent frames.

Besides providing significant improvement in move-stop-move tracking, feature-based target position measurements are helpful for tracking closely-spaced targets. As shown in Fig. 13(a), two sets of three vehicles pass each other with very little separation, leading to partial occlusion of the targets. When in close proximity, a single motion-based measurement may be obtained for two vehicles, but feature-based measurements can be obtained for both vehicles. Thus, combined with Joint Probabilistic Data Association, feature-based measurements can provide significant improvements in track continuity for closely spaced targets, as illustrated in Fig. 13(b), where the target tracks have separated correctly. The image plane tracker continues to maintain track purity throughout the remaining challenging scenarios presented by the VIVID EgTest02 video.



**Figure 13:** In the EgTest02 video, two sets of three vehicles pass each other with very little separation, but are tracked effectively via a combination of motion- and feature-based measurements with JPDA.

## 6. CONCLUSIONS

In this paper we have described the algorithms upon which the image plane tracker is based. Results presented for the publicly-available Procerus™ UAV Convoy Following and VIVID EgTest02 videos have demonstrated the effectiveness of the IPT in tracking multiple closely-spaced moving targets in the presence of significant platform motion and measurement clutter, in scenarios which are representative of real-world challenges faced by video exploitation software. Thus, we believe the image plane tracker can serve as a valuable tool for assisting pilots/operators and commanders in effectively utilizing the available video sensor platforms and achieving mission effectiveness.

## ACKNOWLEDGEMENTS

The authors thank Todd V. Rovito and AFRL/SNAT for providing helpful suggestions and supporting this research.

## REFERENCES

- <sup>1</sup> Trick R/C™, <http://www.zagi.com/>
- <sup>2</sup> Procerus™ Technologies, “UAV Convoy Following,” <http://procerusuav.com/gallery.php>
- <sup>3</sup> Robert T. Collins, Martial Hebert, Hulya Yalcin, David Tolliver, Marius Leordeanu, Xuhui Zhou, and Seng Keat Teh, “VIVID Tracking Evaluation Web Site,” <http://www.vividevaluation.ri.cmu.edu/>
- <sup>4</sup> Bruce D. Lucas and Takeo Kanade, “An Iterative Image Registration Technique with an Application to Stereo Vision,” *International Joint Conference on Artificial Intelligence*, pages 674-679, 1981.
- <sup>5</sup> Carlo Tomasi and Takeo Kanade, “Detection and Tracking of Point Features,” *Carnegie Mellon University Technical Report CMU-CS-91-132*, April 1991.
- <sup>6</sup> Jianbo Shi and Carlo Tomasi, “Good Features to Track,” *IEEE Conference on Computer Vision and Pattern Recognition*, pages 593-600, 1994.
- <sup>7</sup> Stan Birchfield, “KLT: An Implementation of the Kanade-Lucas-Tomasi Feature Tracker,” <http://www.ces.clemson.edu/~stb/klt/>
- <sup>8</sup> Robert T. Collins, Xuhui Zhou, and Seng Keat Teh, “An Open Source Tracking Testbed and Evaluation Web Site,” *IEEE International Workshop on Performance Evaluation of Tracking and Surveillance (PETS 2005)*, January, 2005.
- <sup>9</sup> M. Fischler and R. Bolles, “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” *Comm. of the ACM*, Vol 24, 1981, pp. 381-395.

---

<sup>10</sup> Yaakov Bar-Shalom, X. Rong Li, and Thiagalingam Kirubarajan, "Estimation with Applications to Tracking and Navigation," Wiley-Interscience, 2001.

<sup>11</sup> Richard J. Radke, Srinivas Andra, Omar Al-Kofahi, and Badrinath Roysam, "Image Change Detection Algorithms: A Systematic Survey," *IEEE Trans. Image Processing*, v. 14, no. 3, March, 2005.

<sup>12</sup> Chris Stauffer and W. E. L. Grimson, "Adaptive mixture models for real time tracking," *Proc. CVPR* 1999.

<sup>13</sup> P. Kaew/TraKulPong and R. Bowden, "An Improved Adaptive Background Mixture Model for Realtime Tracking with Shadow Detection," *Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems—AVBS01*, Sept. 2001.

<sup>14</sup> Yaakov Bar-Shalom and Thomas E. Fortmann, "Tracking and Data Association," Academic Press, 1988.

<sup>15</sup> Craig Agate and Kevin Sullivan, "Signature-Aided Tracking Using Association Hypotheses" Proceedings of SPIE AeroSense 2002.

<sup>16</sup> David Beckman, Kevin Sullivan, and Craig Agate, "Continuous Identification of Ground Targets Using HRRGMTI Measurements", Toyon Research Corporation, August 2003.