Map-based Active Leader-Follower Surveillance System

Himaanshu Gupta, Xiaochun Cao, Niels Haering

Map-based Active Leader-Follower Surveillance System

Himaanshu Gupta[1], Xiaochun Cao[1,2], Niels Haering[1]

1: ObjectVideo, Inc., 11600 Sunrise Valley Dr., Reston, VA 20191, USA
2: School of Comp. Science and Tech., Tianjin University, Tianjin 300072, China
{hgupta, xcao, nhaering}@objectvideo.com

Abstract. We propose a generic framework for an active leader-follower surveillance system. The system can ingest inputs from a variety of multi-modal leader sensors: Radars, Automatic Identification Systems (AIS), or cameras. Rule and learning based pattern recognition techniques are performed on the leader sensors to infer unusual behaviors and events. In order to calibrate the follower PTZ cameras to the leader sensors, we break up the calibration process into two independent steps: mapping leader cameras' image coordinates to latitude/longitude values, and mapping the latitude/longitude values to the follower's pan-tilt-zoom settings. This calibration method provides straightforward scalability to complex camera networks since newly added cameras need to be calibrated only with respect to the site map (latitude/longitude), and the Radar or AIS leader sensors also fit in seamlessly. Upon detection of an unusual event by one of the leader sensors, follower PTZ cameras move to the location where the event was triggered, detect the target of interest in the field of view, and then track the target actively by automatically adjusting the PTZ settings. We also propose a system to learn various PTZ camera characteristics which facilitate stable and accurate control of the camera during active tracking. Finally, we evaluate the performance of each of the components individually, as well as that of the entire system.

1 Introduction

A leader-follower system is an effective and commonly used solution for wide area surveillance. In such a system, a fixed wide FOV sensor acts as the leader, and directs follower PTZ cameras to zoom in on the targets of interest. One of the drawbacks of a typical active leader-follower system is that the follower camera can only follow the target as long as it remains in the leader's FOV. Also, due to inaccuracies in the leader-follower calibration, the follower camera might be unable to zoom-in to tightly on the target. To automate the follower cameras, we add active tracking capability. There are two challenges faced by the follower camera: 1) update the pan/tilt angles to follow it over long durations, and 2) automatically refine the zoom levels for the optimal tracking and visualization

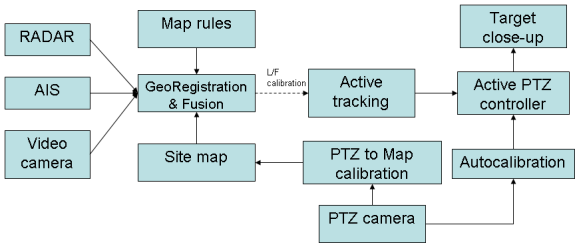


Fig. 1. Active leader-follower surveillance system.

performance, i.e. as big as possible without losing targets. During this following process, the intrinsic PTZ camera properties such as actuation/query delays and pan/tilt/zoom velocities significantly affect the performance. However, deployed video monitoring systems are likely to interact with a wide range of COTS follower PTZ cameras in various states of working order, which renders manual calibration or the use of manufacturer specifications impractical. We thus propose a new method for automatically acquiring the PTZ parameters of a PTZ camera by sending sequences of commands to the camera and analyzing the incoming video. Several methods [1–4] are proposed to calibrate pan-tilt cameras. Davis and Chen [1] utilize a moving LED as the virtual landmark. But they require multiple calibrated stationary cameras to infer the 3D locations via stereo triangulation. Zhou et al. [2] builds a look up table for each pixel coordinate in the leader camera and the pan-tilt angles in the follower camera. This method has the disadvantage that the calibration time will increase quadratically with respect to the number of leader and follower cameras. Jain et al. [3] focuses on the close views where the displacement between rotation axis and the optical center matters. Sankaranarayanan and Davis [4] focus on top view cameras and introduce the elliptical locus for the inferring of pan-tilt orientations. Our united leader/follower calibration method has two independent steps: mapping leader camera image coordinates to the latitude and longitude, and mapping the latitude and longitude to follower pan-tilt-zoom values. Our approach is related to [2]. However, the calibration for each new leader and follower is independent and the system can be expanded to n leader and follower cameras in $O(n)$ calibration time, comparing to $O(n^2)$ using [2]. In addition, other leader sensors such as Radar fit in seamlessly since they provide latitude/longitude data directly. Another major component in an active leader follower camera system is online pattern recognition in the leader cameras. The leader camera needs to perform real-time detecting, tracking, classifying and monitoring of suspicious targets, and automatic analysis of their behaviors. We improved the traditional background subtraction method [5] to detect targets in complex environment. Temporal data association for the detected targets is performed using both motion and appearance properties of targets to solve occlusions and handle detection errors and sensor noise. The rule and learning based pattern recognition algorithms are applied to the target physical information after geo-registration. In summary, our leader and follower system is illustrated in Figure 1. These cameras are all

geo-registered into a common world coordinate system as described in Section 3. In Section 2, rule and learning based pattern recognition techniques are applied to the physical properties of geo-registered targets and generate alerts. The alerts come with latitude and longitude that can be converted to pan-tilt-zoom values of a follower camera after the leader/follower calibration in Section 3. The follower PTZ cameras' characteristics, including pan/tilt/zoom profiles and actuation/query delays, are thoroughly analyzed in Section 4. After automatically zooming into the target of interest, an active tracking algorithm is used in Section 5 to follow the target. We present experimental results in Section 7.

2 Leader Sensor Processing

In order to trigger the follower to track the targets of interest, the leader camera needs to be able to detect those targets. For detection in a stationary camera, we improved the subtraction method [5] using a multi-modal spatiotemporal representation of the background to detect targets in dynamic scenes, e.g. water. Temporal data association for the detected targets is performed using both motion and appearance properties of targets to handle occlusions, detection errors and sensor noises. The tracked targets are then classified into predefined classes including humans, groups of humans, vehicles, watercraft (and its subclasses: freighters, cruise ships, recreational boats, etc.) using Gentle AdaBoost [6]. The classifier uses multiple cues that include C2 Features [7], metric size, metric velocity, aspect ratio, and any other available contextual information. Pattern recognition is performed in both the image and world coordinate systems. While the methods for both coordinate systems are essentially similar, in this work we focus on the world coordinate system. Given detected targets with classification types from the stationary cameras, their physical information is computed based on geo-registration results as described in Section 3.1. This physical information is defined in a unified coordinate system, i.e. the world. Then a data fusion service links disconnected target trajectories inside a sensor and observations across sensors in a probability framework by modeling a target's geodesic location, speed and heading. Global consistency is enforced by temporally smoothing target locations and removing spurious targets. We use tripwire and area of interest (AOI) rules specified by the users to generate alerts. We also learn historical target properties for the detection of unusual events. Target properties include target width, height, size and speed. In our system, we learn the boat speed and density information based on the Automatic Identify System (AIS) readings. For each map pixel, we fit a Gaussian model for the normal speed. Therefore, given the estimated target physical speed on one location, we decide whether they are consistent with the historical observation. Radar and AIS data can be seamlessly integrated into this framework and treated as leader sensors. For example, the Radar readings have geo-location and speed information. Geo-location data can be used to trigger map rule based alerts. Together with geo-location data, speed information can be used to compare against historical target speed maps.

3 Leader-Follower Calibration

We propose a camera calibration scheme which involves registering the leader camera to a world coordinate system, where each point is mapped to unique pan-tilt-zoom values of each follower camera. We break this calibration scheme into two steps: geo-registration of leader camera, and calibration of PTZ camera to geo-locations. This two step process allows a flexible approach as far as integrating various kinds of leader sensors (Radar/AIS/camera) in the framework as the second step remains the same.

3.1 Georegistration of Static Cameras

We use a planar homography to map the camera field of view to a satellite image/map that has pixel level geo-registration, e.g. Google Earth [8]. This relationship is estimated using geo-positioning information broadcasted by the mobile transponders as detailed in [9]. The transponders can be persons or cars on land, and boats on water. The footprints of the transponders in camera views are automatically detected and tracked. Note that in an uncontrolled environment, neither do all detected targets in the camera view transmit geo-positional data nor do all targets transmitting this data have to be visible. Therefore, one-to-one correspondences between target footprints and geo-positional data are not available. In our experiments, accidental correspondences (noise) significantly out-number the true correspondences (typically noisy data is more than 80% of the total data), where a RANSAC framework [10] is not guaranteed to find an optimal homography. To handle this challenge, aggressive data filtering is first utilized to reject outliers based on motion properties, lengths and confidence of trajectories, global consistency of motion, as well as geometrical constraints.

3.2 Map to PTZ Camera Calibration

During the calibration process, users manually slew the follower camera to a landmark at $\mathbf{P}_W = [Latitude, Longitude]^T$, and record the pan and tilt angles as θ and ψ. The two readings θ and ψ define a 3D ray casting from the center of the follower camera's pan-tilt head. This ray intersects a virtual plane $\mathbf{\Pi}$, which is parallel to the camera's mount plane at unit distance, at location $\mathbf{P}_v = [x_v, y_v]^T$ as:

$$x_v = \tan(\theta)\sin(\psi), \tag{1}$$

$$y_v = \tan(\theta)\cos(\psi). \tag{2}$$

Given more than four non-degenerate landmarks and corresponding pan and tilt readings, we find the planar homography $\mathbf{H}$ [10] that projects the ground plane points $\mathbf{P}_W$ to points $\mathbf{P}_v$ on the virtual plane $\mathbf{\Pi}$, which provides θ and ψ after triangular operations. This completes the calibration for the pan and tilt values. The optimality of zoom is subjective and varies from applications. While users would like to zoom to the faces of a detected person as much as possible, a fast moving person might go out of field of view due to the detecting, panning

and responding latencies. To approximate the user's desire for a particular zoom function, then, the calibration procedure fits a least-squares line through the zoom values given by the user based on the calculated distance from the target to the camera location on the world plane

4 PTZ Camera Autocalibration

As in any control system, visual feedback (as opposed to open-loop control) enables an active PTZ controller to overcome calibration and modeling errors when tracking moving targets. Calibration of the PTZ head is an important step in implementing active control, in particular for tuning the response and stability of the controller. Deployed systems are likely to interact with a wide range of COTS PTZ cameras in various state of working order, which renders manual calibration or the use of manufacturer specifications impractical. As a result, we have developed calibration techniques to automatically acquire the parameters of the PTZ camera by sending sequences of PTZ commands and analyzing the resulting video.

4.1 Focal Length and Zoom Profile

Many COTS cameras implement an arbitrary non-linear scale between zoom commands and actual magnification. Knowledge of this zoom profile is required for the zoom controller to maintain a visual target at an optimal size on the image plane. Furthermore, knowledge of the focal length at each zoom level allows pixel offsets to be converted into pan and tilt angles for active camera control. We model the zoom profile as a high dimensional polygon relating zoom to magnification. As the first step in this calibration process, the camera is set to minimum zoom and commanded to follow a short scan path over a pan range of +/-0.5 radians. Captured frames are processed using the usual mosaic building algorithms. After the first complete scan is detected, the mosaic optimization algorithms [11] are applied to estimate the focal length. The estimated focal length is recorded as the base value to which the zoom profile is applied to obtain the focal length at a given magnification. To estimate the zoom profile, a series of zoom commands over the full working range are sent to the camera, and the frame before and after execution of each command is logged. The magnification between each pair of frames is then calculated using a brute force search over scales. Multiplying the scales over all frame pairs yields the complete zoom profile. Finally, a high dimensional polygon (of order 10 in the current implementation) is fitted to the measurements to model the zoom profile.

4.2 Pan and Tilt Velocity Profiles

The pan and tilt velocity profiles characterize the relationship between the commanded pan/tilt velocity and the actual velocity, along with the maximum physical velocity for each axis. The profile is represented by fitting a high order polynomial to measured velocities uniformly spaced over the input velocity range.

For each measurement, the camera is driven at the fixed input speed starting from the minimum pan/tilt range for a fixed interval T, and the PTZ position is queried before and after each motion. The measurement is then repeated from the minimum pan/tilt angle for an interval of $2T$. The angular difference between the two motions divided by T gives the measured velocity for the given input. Measuring the velocity this way ensures that errors due to actuation delays and acceleration of the axes are cancelled out when taking the difference. The maximum pan/tilt velocity is the highest input setting that results in a unique velocity measurement.

4.3 Zoom Position and Duration Profiles

While the pan and tilt velocities are typically independent of the camera position, the zoom velocity varies with zoom setting. Thus, we can not represent the zoom profile as a simple velocity input/output relationship. Instead, we construct a zoom position profile which gives the zoom setting attained after zooming in from the minimum zoom for a given duration at a given speed setting. We also represent the inverse of this curve, the zoom duration profile to simplify computation during velocity control of the camera. In theory, the zoom profiles should be measured for each zoom speed setting of the camera; in practice, we always use the minimum zoom speed setting and therefore only measure one profile pair. The profiles are represented as high order polynomials fitted to a set of measurements, where each measurement (T, z) is obtained by zooming the camera from the minimum zoom for duration T and querying the final zoom z.

4.4 Minimum Query Interval and Query Delay

A query command returns the current location of the pan, tilt and zoom axes from the PTZ controller. The minimum query interval is the minimum time between sending consecutive query commands, and the query delay is the duration between sending a query command and receiving a response from the controller. In the best case, the minimum query interval is equal to the query delay. The minimum query interval is measured by issuing pairs of query commands temporally separated by increasing multiples of 100 ms and taking the smallest interval for which the second query returns a valid response. The query delay is estimated as the difference in system time between sending a query and receiving a response, averaged over several commands.

4.5 Actuation Delay

The actuation delay is the time between issuing a velocity command and observing the resulting camera motion, and results from signal processing delays and mechanical characteristics such as acceleration. To determine when the camera starts moving, each captured frame is classified as moving or stationary based on frame differencing and thresholding with the previous frame. The actuation delay is then measured as the difference in system time between sending a motion command and observing the first moving frame, averaged over several motion

commands. The actuation delay is measured separately for the pan/tilt and zoom axes, since these use different physical mechanisms.

5 Active Tracking

Once the PTZ camera is pointed in the vicinity of the target of interest using the leader-follower calibration described above, the tracking algorithm assumes the task of segmenting the target and estimating the state (position, orientation, scale and velocity) in subsequent frames. Our tracking algorithm exploits the fact that watercraft are generally sparsely distributed. This enables the effective use of adaptively selected discriminative features [12][13] and tracking of targets independently without explicit consideration of multiple target tracking. New targets are initialized from the detection using the same algorithm described in Section 2 that do not overlap with existing targets. Our tracking algorithm uses adaptively selected discriminative color subspace features based on the approach in [12]. A discriminative subspace provides good separation between foreground and background pixels; the tracking algorithm evaluates the most discriminative subspaces in each frame and uses those to track the target in the subsequent frame. During tracking, the current frame is projected into each color subspace. A foreground histogram ϕ_{fg} is accumulated from pixels in the target ellipse and a background histogram ϕ_{bg} is accumulated from pixels in a surrounding elliptical ring. A log likelihood ratio is then computed for each histogram bin as:

$$L(i) = \frac{\max\left(\phi_{fg}(i), \delta\right)}{\max\left(\phi_{bg}(i), \delta\right)} \tag{3}$$

where $\delta = 1/\max\left(N_{fg}, N_{bg}\right)$, and N_{fg}, N_{bg} are the number of foreground and background pixels respectively. Finally, a likelihood image is constructed by projecting each pixel into the N_f most discriminative color sub-spaces from the previous frame and averaging the likelihood ratio for the corresponding histogram bins. Target pixels in the resulting image are expected to have a positive likelihood while background pixels have a negative likelihood. The discriminability of each color subspace is measured after updating the target in each frame, and the best N_f subspaces are used to track the target in the subsequent frame. Since evaluating multiple subspaces is computationally expensive, two measures are implemented to maintain real-time tracking performance. Firstly, the likelihood map is only computed in a local region around the predicted target location. Furthermore, the image is dynamically sub-sampled to maintain an approximately constant number of pixels on target over all scales. A mean-shift procedure [14] is used to update the location and scale of the target given the likelihood image. Before applying mean-shift, the target location is predicted from the previous frame using the estimated velocity and inter-frame sample time. Targets can undergo a significant change in scale while moving from far field to near field in the camera field of view. To maintain track, we use two inter-leaved mean-shift searches over location and scale respectively, similar to [14]. Finally, a Kalman Filter is applied to the new target location from the mean-shift procedure to estimate the location and velocity of the target ellipse.

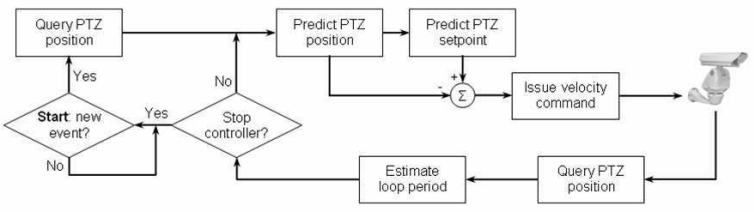


Fig. 2. Active tracking controller

6 Active PTZ Controller

Most of the standard PTZ control algorithms issue absolute pan, tilt, and zoom commands to the camera. While this approach is sufficient for tracking and maintaining the target in the camera FOV, it leads to jerky camera motion. In order to improve the viewing experience for the user by having the camera move smoothly and continuously, we implemented a velocity-based active PTZ controller that issues velocity commands rather than position commands. Furthermore, increments in velocity are typically less perceptible than increments in position. Figure 2 provides an illustration of the active PTZ control loop. While no targets are visible, the controller remains in a wait state. When a new active tracking event occurs, the controller queries the current location of the PTZ and enters the main control loop. The first step in the control loop is to predict the change in camera position (due to previously issued velocity commands) between the time at which the frame was processed and the time when the camera would execute the command. The change in camera location is predicted by adding the integral of all velocity commands issued between the time when the last video frame was processed and the time of command execution (current time + actuation delay). The next step is to determine the desired pan, tilt, zoom correction of the camera in order to center the target at the desired resolution. The system computes the camera zoom level at the time of frame processing by aggregating all the zoom commands issued since the last camera position query up to the time at which the last frame was processed to the last queried zoom level. Once the zoom level is known, using the zoom profile(Section 4.1), the focal length of the camera at that particular frame can be determined. The pan and tilt correction can then be computed as follows:

$$\triangle \text{pan} = atan(\frac{\text{frame\_center}_x - \text{target\_center}_x}{\text{focal\_length}}) \tag{4}$$

$$\triangle \text{tilt} = atan(\frac{\text{frame\_center}_y - \text{target\_center}_y}{\text{focal\_length}}) \tag{5}$$

The PTZ set point is the desired positional correction of the camera, and is computed by extrapolating the last known location of the tracked target. The target velocity is estimated using the previous two target centering differentials and the aggregate camera motion between the previous two target sightings. This

velocity is integrated over the time difference between the time of last frame processing and the time when we anticipate the target to be centered in the camera FOV (current time + actuation delay + target centering duration). The difference between the predicted PTZ position ($\boldsymbol{p}$) and predicted setpoint ($\boldsymbol{s}$) serves as the control loop error signal from which a proportional velocity command ($\boldsymbol{v}$) is computed. The controller gain is determined indirectly by a parameter called the target centering duration (d), which is the duration required for the camera to move to the current setpoint. The controller gain, in combination with the control loop period τ, has a significant impact on the stability of the controller. Since the period of the control loop is not known ahead of time, we estimate it online and compute the centering duration as a factor (f) of the control loop period. Thus, the new velocity command is computed as:

$$\mathbf{v} = \frac{\mathbf{s} - \mathbf{p}}{d} = \frac{\mathbf{s} - \mathbf{p}}{f\tau} \tag{6}$$

The pan and tilt velocity commands issued to the camera are computed using the pan/tilt velocity profiles computed in 4.2. The duration of the zoom command is determined by the required zoom correction and the zoom duration profile (4.3). After issuing a new velocity command, the camera is queried for its current position. If the query delay is significant, the control loop can be configured to query less frequently than every cycle in order to reduce the control loop period and improve controller stability. The control loop period for the current loop is estimated using standard system timers, and is combined using a fading memory average with previous estimates.

7 Experimental Results

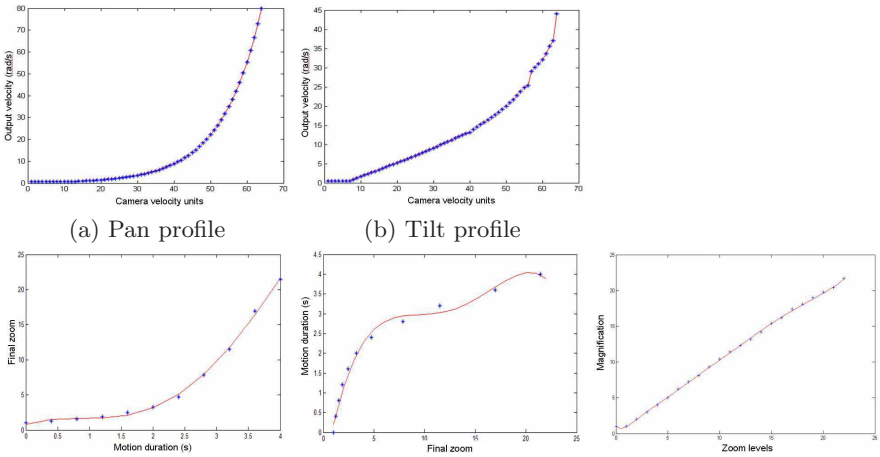
In this section, we present the experimental results of the various technologies that we have proposed in the paper so far.

7.1 PTZ Autocalibration

Once the PTZ autocalibration process is complete, the system has knowledge about the various characteristics of the camera that are essential for accurate control during active tracking. We used a COTS Pelco Spectra III camera for our experiments. Figures 3(a)&(b) depict the mapping between input pan and tilt velocity in camera units and the output velocity in radians/sec. Figures 3(c)&(d) depict the zoom position and duration profiles respectively. Figure 3(e) depicts the mapping of camera zoom levels to image magnification. For the Pelco Spectra III under test, the minimum query interval was 500 ms and the query delay was about 400 ms. The estimated pan/tilt and zoom actuation delays were 70ms and 130ms respectively.

7.2 PTZ to Map calibration

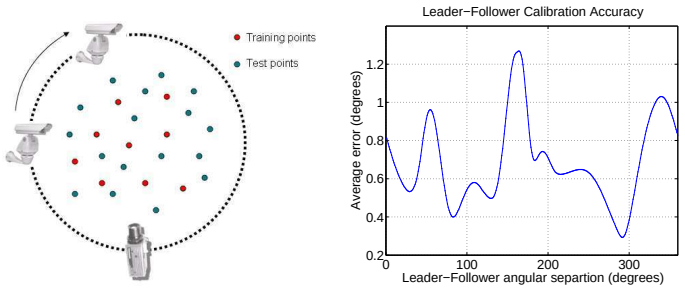
To test the accuracy of our proposed calibration algorithm, we performed a qualitative and quantitative analysis to gauge the performance. In a real world



(a) Pan profile (b) Tilt profile

(c) Zoom position profile (d) Zoom duration profile e) Zoom Profile

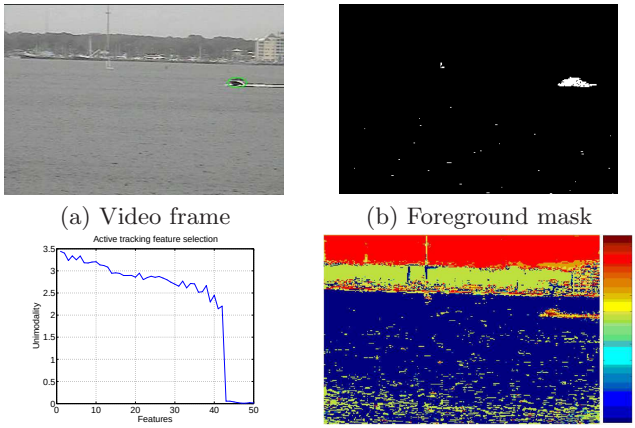
Fig. 3. PTZ Velocity Profiles



(a) Setup (b) Pan-Tilt accuracy

Fig. 4. Leader-follower calibration accuracy

setting, the camera at our test site was calibrated to the map, and upon clicking at a point on the map, the camera was commanded to go to the corresponding pan, tilt and zoom values (computed using the calibration). A human observer then verified if the camera was pointing at the correct location. We used synthetic data to measure the quantitative performance of the system. Figure 4(a) shows a pictorial representation of setup used for the synthetic data analysis. We used 8 training points (red) to compute the calibration matrix, and used 300 test points (green). The PTZ camera was moved in a circle around the points to measure the effect of the camera viewing angle on the calibration accuracy. We added a uniform noise of 1 pixel to take into account typical operator error during the calibration process. Figure 4 (b) shows the plot of the resulting average pan/tilt error versus the PTZ camera viewing angle. As can be seen from the plot, the camera positioning is accurate up to a degree for almost all viewing angles.



(a) Video frame (b) Foreground mask

(c) Feature discrimination score (d) Likelihood map

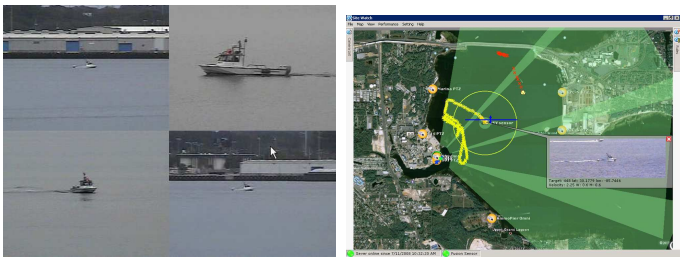
Fig. 5. Active tracking results

7.3 Active Tracking

Figure 5 illustrates the results of some the intermediate stages of the active tracking algorithm. Figure 5(a) is the video frame from the PTZ camera after it has been pointed in the target's vicinity using the map-PTZ camera calibration. Figure 5(b) shows the foreground mask detected using the algorithm described in Section 2. Figure 5(c) shows the sorted feature discrimination score for the candidate color subspaces. Figure 5(d) shows the averaged likelihood map image computed using the two features with the greatest feature discrimination score. with the maximum discrimination score. The detected target ellipse computed after performing mean-shift iterations on the likelihood image is overlayed on the original video frame and shown in Figure 5(a). Figure 6(a) shows the snapshots of four PTZ camera at the test site while tracking the same target. Note that some cameras have less than desired resolution on the target since they are already fully zoomed in.

7.4 Event Detection

The event detection performance was also evaluated on the ground truth data. In this evaluation, we defined a virtual tripwire as the event trigger. Tripwires were randomly drawn and the ground truth for each tripwire crossing was manually labeled. Figure 6(b) shows a boat crossing one such tripwire along with the mapped target trajectory and target snapshot. The hit/miss/false alarm rates for seven test video clips were measured. Overall, the system was able to detect 55 out of 56 tripwire events with a total of 7 false alarms. The precision and recall for event detection is 88.71% and 98.21% respectively.



(a) (b)

Fig. 6. (a) Active tracking PTZ snapshots (b) Estimated FOVs of four high resolution cameras overlaid on satellite imagery, along with map rules and target trajectories.

References

1. Davis, J., Chen, X.: Calibrating pan-tilt cameras in widearea surveillance networks. In: Proc. IEEE ICCV. (2003) 144–150
2. Zhou, X., Collins, R.T., Kanade, T., Metes, P.: A masterslave system to acquire biometric imagery of humans at distance. In: Proc. ACM SIGMM international workshop on Video surveillance. (2003) 113–120
3. Jain, A., Kopell, D., Kakligian, K., Wang, Y.: Using stationary-dynamic camera assemblies for wide-area video surveillance and selective attention. In: Proc. IEEE CVPR. (2006) 537–544
4. Sankaranarayanan, K., Davis, J.: An efficient active camera model for video surveillance. In: Proc. IEEE WACV. (2008)
5. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and practice of background maintenance. In: Proc. IEEE International Conference on Computer Vision. (1999) 255–261
6. Friedman, J., Hastie, T., Tibshirani, R.: Additive logistic regression: a statistical view of boosting. Ann. Statist. **28** (2000) 337–407
7. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Robust object recognition with cortex-like mechanisms. IEEE Transaction on Pattern Analysis and Machine Intelligence **29** (2007) 411–426
8. : (http://earth.google.com/)
9. Haering, N., Javed, O., Rasheed, Z., Shafique, K., Cao, X., Liu, H., Yu, L., Madden, D., Chosak, A., Taylor, G., Gupta, H., Lipton, A.: Automatic imaging sensor calibration using objects that broadcast positional information. US Patent Application, # 37112-230561 (2007)
10. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
11. : A framework for wide-area visual surveillance using a scanning camera. (Under review (please see attached supplemental material))
12. Collins, R., Liu, Y., Leordeanu, M.: On-line selection of discriminative tracking features. IEEE Transaction on PAMI **27** (2005) 1631–1643
13. Han, B., Davis, L.: Object tracking by adaptive features extraction. International Conference on Image Processing **3** (2004) 1501–1504
14. Collins, R.: Mean-shift blob tracking through scale space. IEEE Conference on CVPR **2** (2003) 234–240