Real-time automated detection, tracking, classification, and geolocation of dismounts using EO and IR FMV

J. Muncaster*, G. Collins, J. Waltman
Toyon Research Corporation, 6800 Cortona Drive, Goleta, CA, USA 93117

ABSTRACT

The VideoPlus®-Aware (VPA) system enables autonomous video-based target detection, tracking and classification. The system stabilizes video and operates completely autonomously. A statistical background model enables robust acquisition of moving targets, while stopped targets are tracked using feature-based detectors. An ensemble classifier is trained for automated detection and classification of dismounts (i.e., humans) and a planar scene model is used to both improve system performance and reduce false positives. A formal evaluation of the VPA system was performed by the government, to quantify the system's abilities to detect, track, and classify, humans. The evaluation provided 811 separate data points gathered over a period of four days with an overall probability of sensing of 99.9%. The probability of detection was 86.2% and the percentage of correct action classification was 82%. The data provided a False Alarm Rate of 0 per hour and Nuisance Alarm Rate of 0.72 per hour. Dismounts were reliably classified with pixel heights as low as 25 pixels. Real-time automated detection, tracking, and classification of targets with low false positive rates was achieved, even with few pixels on target. The planar scene model based optimizations were sufficient to dramatically reduce the runtime of sliding-window classifiers.

Keywords: Automated Detection, Tracking, Classification, Video Surveillance, Video Analytics, Automation

1. INTRODUCTION

Traditional surveillance applications rely heavily on a human operator to detect suspicious activity. In recent years automated processing has been introduced into surveillance systems with limited success. The field of *video processing* or *video analytics* attempts to automate the detection, tracking, and classification of specific objects and/or actions as they appear in video, as shown in Figure 1. This is accomplished using specialized algorithms and software that analyze a video stream and search for specific shapes, motions, or features. Some discussion of this technology appears in [1]-[4], and the references therein. Video analytics have made substantial progress in the past decade. Technologies now exist which can reliably detect and track moving targets in video. More recent work has also made progress on automatically classifying specific objects and actions. However, a persistent problem faced by all of the aforementioned technologies is that of *false alarms* and *nuisance alarms*. These terms refer to detections or alarms generated by the video analytics system on objects/actions that are not of interest (formal definitions for these terms are provided in Section 3). In most current implementations of video analytics technologies, Nuisance Alarm Rates (NARs) are often so high that the operators of these systems lose trust in them, causing many operators to disable or ignore the outputs of such systems.

Most current systems take a uni-modal approach to detecting intruders. Many rely solely on motion while making unrealistic stationary-background assumptions. This leads to unacceptable NAR rates when the camera is buffeted by wind or in the presence of natural scene variations such as changes in lightning, sensor auto-gain, blowing foliage, sensor noise, or water motion. Other approaches rely only on global sliding-window classifiers, which can produce unacceptable false alarm rates on complex backgrounds.

Toyon Research Corporation has addressed this problem by taking a multi-modal approach to detecting and tracking targets. By combining several modes of detection, we have built a system that is more robust to environmental conditions which typically cause nuisance alarms. Toyon's approach is to stabilize the image and build a statistical background for motion-detection. A short calibration procedure builds a planar scene model, allowing incorrectly sized detections to be filtered out. A multi-target tracker instantiates target tracks and predicts their locations in subsequent frames. A feature model is learned for each track, and localized searches enable the system to lock on to stopped targets. Finally, a sliding-window classifier performs local searches using the predicted target location and the planar model of

*jmuncaster@toyon.com

target size to both detect and classify moving or stationary targets. System components are integrated coherently, producing a system with low false alarm rates and the capability to detect and track targets at long ranges and with low numbers of pixels-on-target. We have combined these technologies into a lightweight network appliance called VideoPlus®-Aware (VPA). Our VPA system is a for site surveillance applications that require exceptionally low false-positive rates.

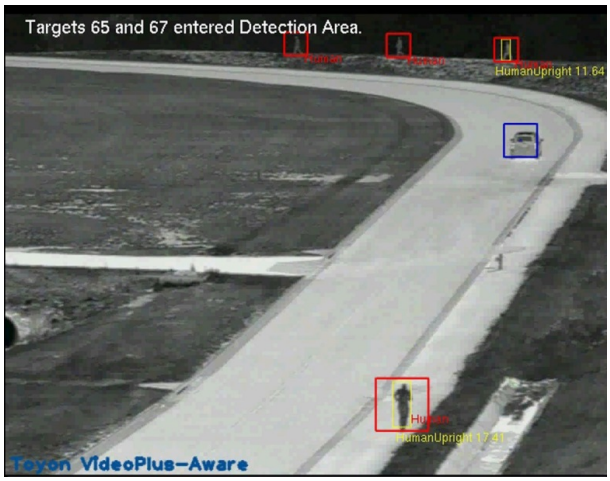

Figure 1. Toyon's VPA technology automatically detects, tracks, and classifies humans in visible and thermal video.

System overview

The VideoPlus®-Aware device and a graphical overview of the surveillance system is shown in Figure 2. The surveillance system consists of one or more cameras, a VPA device, and command and control (C2) software. The VPA device is a small, lightweight computer that runs Toyon's video processing algorithms. The device takes as input digital video received over a standard Ethernet connection. VPA will enhance video quality, automatically detect and track targets in the video, classify targets in video, and provide user-customizable alerts, as shown in Figure 1. VPA provides as output (1) a video stream with annotations indicating the locations of targets in the image, and (2) a meta-data stream that contains the alerts in an open-standards format. VPA is configurable using an HTTP interface, and provides both a native and a web-based GUI. Outputs are consumed by the C2 software and rendered geodetically onto the earth. Special Technologies Laboratory *RaptorX* C2 software was used for the evaluation discussed in this paper.

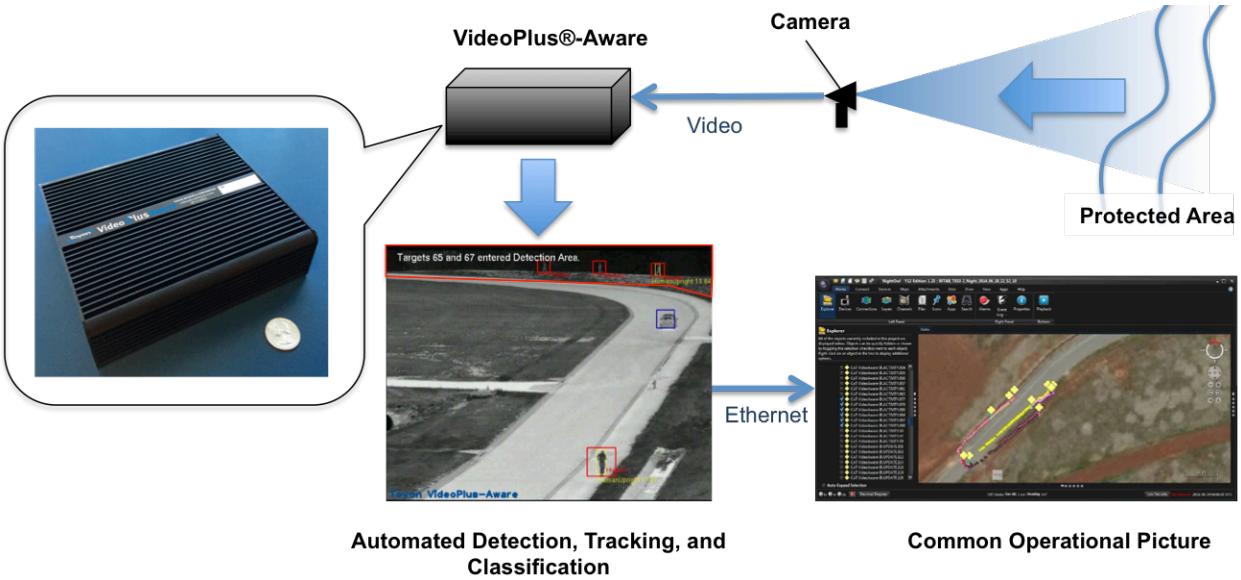

Figure 2. VideoPlus®-Aware system overview and low-SWaP analytics device

The structure of this paper is as follows. We begin in Section 2 where we describe the primary algorithmic components of VPA, with broader discussion of optimizations that enable robust, real-time sliding-window-based detection. In Section 3 we describe the setup of a performance evaluation that was performed. In Section 4 we discuss results, and in Section 5 we end with conclusions and future research goals.

2. ALGORITHMS

Motion video is made up of a sequence of still images, called *frames*. Typical NTSC video runs at 30 frames per second (FPS). Our VPA software operates on the individual frames of the video sequence. An overview of the VPA video processing components is shown in Figure 3. For each video frame, an input image is processed and the relative motion of the camera is estimated[1]. Tracks are maintained in image-plane coordinates and track locations are predicted to each frame, accounting for both the estimated target velocity and the motion of the camera. The camera motion model, environment model, and track predictions are sent along with the image to three complementary detection schemes, to be discussed in subsequent sections. Target detections are then sent to the tracking module, which performs joint probabilistic data association (JPDA) and estimates target image-plane locations and velocities. In addition, target size, feature, and class information is fused into a target model for each track and the track is updated. Finally, the target is projected to geodetic coordinates and activities are recognized based on the target state and estimated class. Track reports and activity reports are finally sent to downstream processes (such as C2 software), based on user-customizable alert settings.

This processing flow creates a chain of serial image processing steps, which are accelerated by pipelining the entire process. This enables efficient use of all processing cores on the host computer. In addition, for applications involving high-resolution imagery, our VPA software has an implementation for a Graphics Processing Unit (GPU), for increased parallelism. Although the VPA unit shown in Figure 2 does not use a GPU, it achieves real-time performance on VGA video with modest hardware.

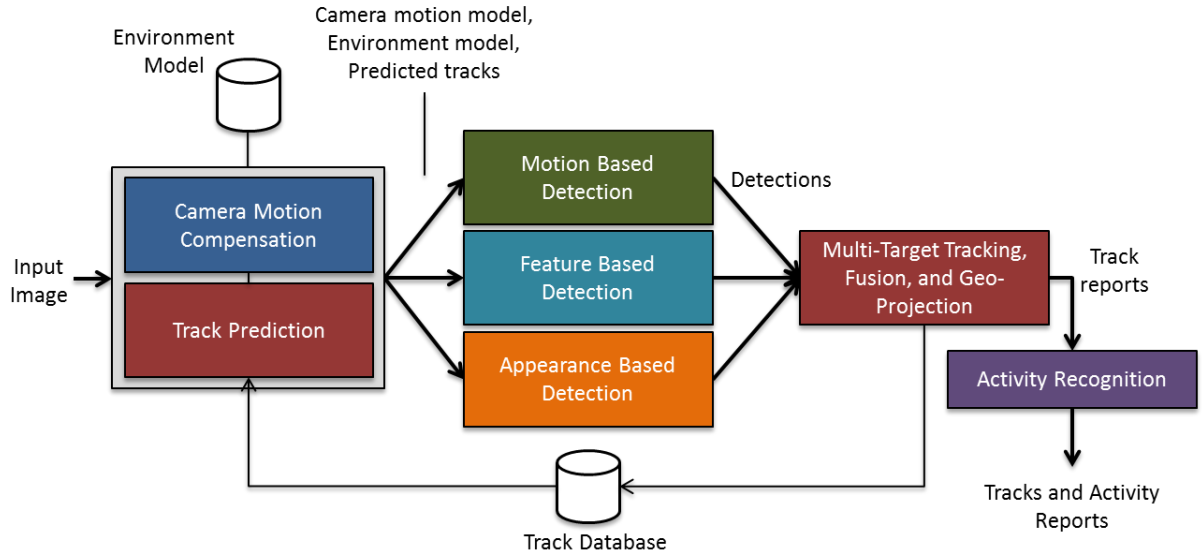

Figure 3. Video Processing System Overview

Calibration and environment modeling

In order to know the typical sizes of targets one must have a model of the 3D scene structure. VPA uses a planar scene model to estimate typical pixel sizes for targets. Configuration of the VPA system includes a step where we calibrate the scene model. This amounts to specifying four calibration points in the scene. For each calibration point, the VPA user

[1] Note that even stationary cameras have levels of jitter that could adversely impact downstream processing if not corrected for.

must estimate the height of an average-sized human (1.68m) standing at that location, as shown in Figure 4. From these sizes, the ground plane is estimated, and the size of a dismounted target at any point in the scene can be estimated using interpolation. This ground plane model is used to calibrate all of the downstream algorithms. The VPA detectors and tracker use typical target sizes and speeds to automatically configure their respective parameters on a per-pixel basis. The motion-detector, for instance, intelligently clusters foreground pixels into smaller groups or larger groups depending on the expected target size at a particular location in the image, while the tracker adjusts its motion-model according to target position so that it models smaller changes in target motion at distant points.

Figure 4. Four virtual humans are sizes coarsely estimated to initialize the planar model. The red box indicates a human that was automatically detected, tracked, and correctly classified.

Camera motion compensation and track prediction

To infer camera motion VPA uses the Kanade-Lucas-Tomasi (KLT) ([5], [6]) feature tracker to locate stable feature points and track them across sequential frames. For every pair of sequential frames, the KLT algorithm locates corner-like feature points that satisfy a constraint on gradient orientations in a local region. These points are efficiently tracked from frame-to-frame using Newton-Raphson iterative approach. A set of feature-point correspondences is collected and used to estimate affine parameters that describe the camera motion model. Random Sample Consensus (RANSAC) algorithm [7] is then used to robustly estimate the camera motion model while accounting for erroneous, outlier feature-point correspondences. The camera motion model is used to convert track locations in a previous image to track locations in the current image. It is also used in the motion-detection algorithms to build a precise background model that is uncorrupted by camera motion.

Moving target detection

To detect moving targets, a background model is created and frame-differences are evaluated against it. We refer to our proprietary background-model as the Typical Apparent Change Model (TACM) [3]. The TACM algorithm performs very well at suppressing localized temporal changes. The TACM algorithm has been applied to a number of environments, including stationary cameras, pole-mounted cameras, and cameras on-board small unmanned aerial vehicles. The primary benefits to the TACM algorithm are that (1) it is able to suppress background variability *without suppressing the foreground target itself*, (2) the background model can be learned in a short amount of time, and (3) it is very computationally efficient. The algorithm works by quickly estimating typical frame-difference variability on a per-pixel basis without corrupting the model by the presence of a target. Subsequent frame-difference are normalized by the variability model. The TACM-normalized change detection metric identifies foreground as outliers in the background-model.

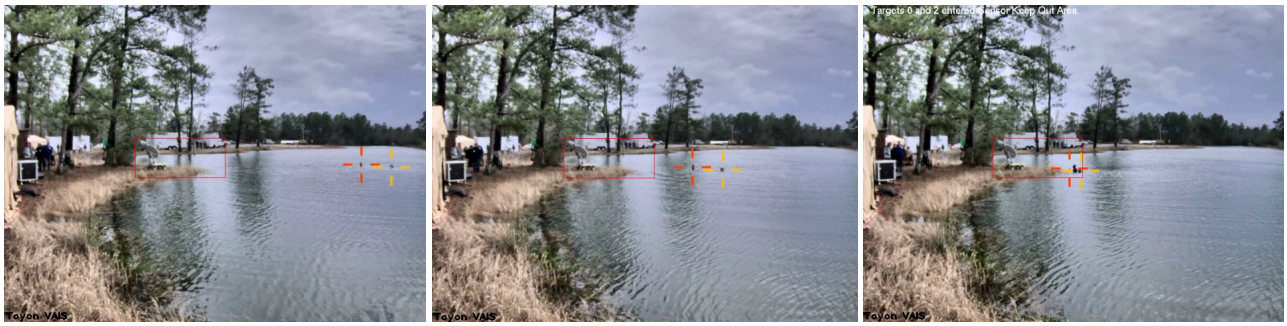

Figure 5. Two amphibious targets are detected and tracked as they enter a protected area.

In Figure 5 we show three frames from the *amphibious assault* video. This video was processed live during an experiment in which a simulated enemy was infiltrating sensitive infrastructure. The targets enter the scene from the right and are automatically detected and tracked as they enter a protected area. This scene contains significant natural motion due to the movement of a large body of water. The effectiveness of the TACM clutter-suppression algorithm is shown in Figure 6. In (a) the difference image shows the high degree of motion from the water movement. In (b) we show a visualization of the TACM background model and in (c) we show our TACM-normalized change-detection metric. Note that virtually all of the motion from the water is suppressed while the motion from the targets is detected. In (d) we show the result of applying a threshold to the change detection metric, and we successfully detect both targets.

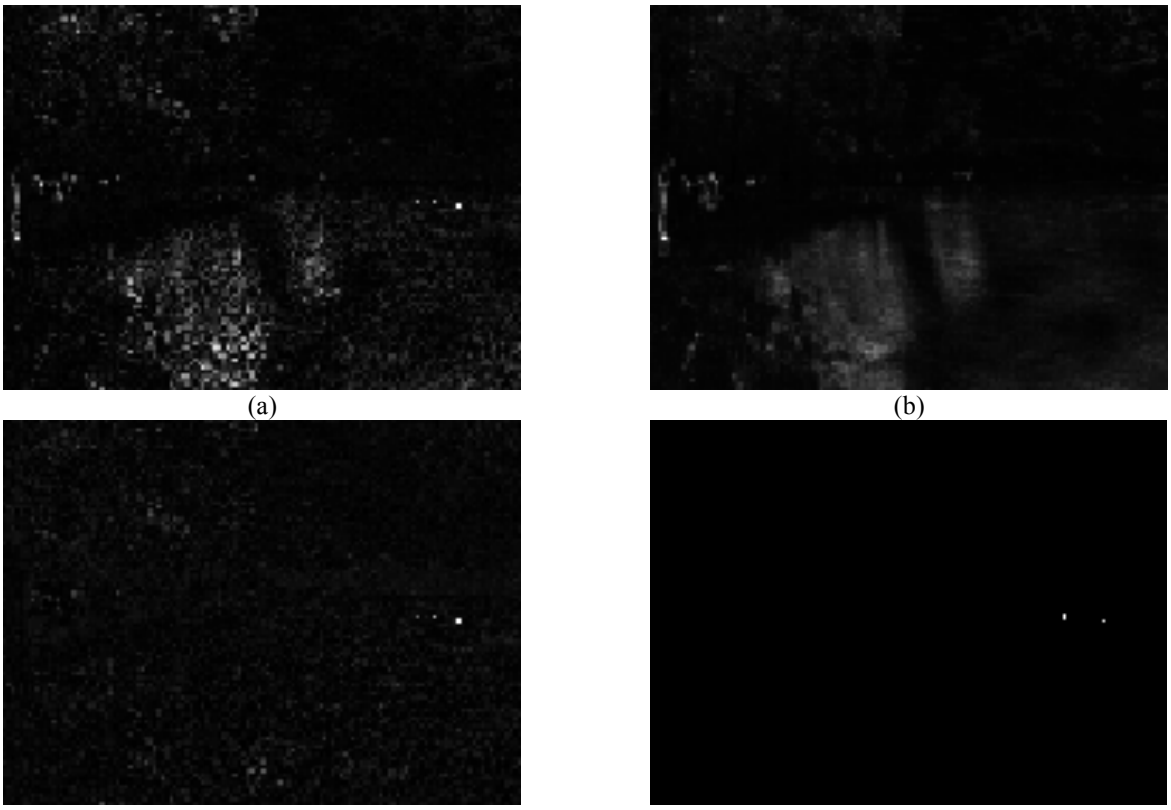

Figure 6: TACM clutter suppression in *amphibious assault* video. In (a) we show a difference image for a scene with a large amount of water movement. Notice the significant apparent movement from the water. In (b) we show a visualization of the background model, where regions with high typical variability are shown in brighter intensity. In (c) we show the clutter suppressed change-detection metric. Note how the water motion has been normalized out but the targets have not. In (d) we show the segmentation mask, detecting both amphibious targets.

Feature based detection

In this section we use the term *feature-based detection* (FBD) to refer to a detection method for detecting targets that are already in track on which we have acquired features characterizing their appearance. Toyon's FBD algorithm learns models of target appearance in on-the-fly in real-time. The algorithm enables both the detection of stopped targets as well as the resolution of closely spaced targets that might become confused in a detection scheme based solely on motion. The structure of Toyon's FBD algorithm is outlined in Figure 7.

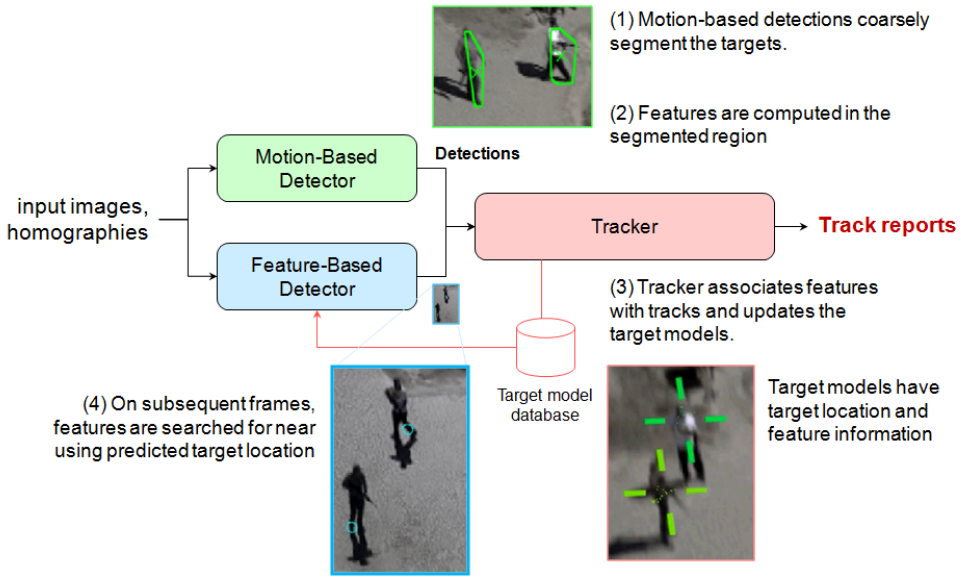

Figure 7: Feature Based Detection Overview

Each time a target is detected using motion, the FBD algorithm computes features that characterize the detection. Toyon's approach uses local image patches as features. Upon the arrival of a series of motion-based detections the tracking algorithm initiates tracks on the moving target. In the process of creating a track the tracker associates motion-based detections with tracks and thus each target has a set of feature points associated with it. On subsequent frames, the feature-based detection algorithm searches for feature points associated with the target in an attempt to locate it in the image frame.

Using the KLT feature-point tracker we implement a feature-based detector as follows. When a new (motion-based) measurement arrives we compute corner-like points to track and associate those feature points with the target. We also take note of the measurement centroid. On subsequent frames, we:

1. Predict the location of each feature point associated with a target and also predict the location of the measurement centroid using the target motion model and the camera motion model.
2. Correct the location of those feature points using KLT sparse optical flow to compute correspondences. Since KLT is a local gradient-descent based method, we seed the KLT algorithm using our predicted feature point locations to reduce the chance of a false match. Next, an affine transformation that describes the feature-point motion is estimated using RANSAC for outlier rejection.
3. Generate a measurement based on the corrected measurement centroid. An affine transformation is used to warp the previous measurement centroid and produce the new measurement location.

Figure 8: Propagated Feature Points for a Target in Track

Finally, as new measurements are associated with the target, the set of feature points associated with the new measurement is used to update the target feature model with additional feature points. We keep only feature points that fall within the three-sigma uncertainty ellipse surrounding the estimated target position to avoid polluting the feature model with points that are on the background rather than on the target. Figure 8 shows a target and its associated feature points.

Target classification using aggregated channel features

In VPA we use the appearance of a typical class of targets to perform simultaneous detection and classification. A *sliding-window* based detector is one that uses local features to determine whether an objects of a particular class is present at a given location. The sliding window is a small rectangular box that is moved around to different locations in the image. Inside the box, our algorithm computes *features* that are unique for objects of different types and then evaluates a classifier designed specifically for detecting targets of a particular type. For this work we were focused on detecting dismounts.

The VPA system uses an extension of the Aggregated Channel Features (ACF) classifier described in [8]. The features are based on 10 channel images, consisting of color (in Luv color-space), gradient magnitude, and a six-bin gradient orientation histogram. A small number of pyramid levels are computed and a dense pyramid is interpolated using the power-law approximation described in [8]. We have found that this detector is among the fastest of those that achieve state-of-the-art performance, consistent with the findings in [4].

ACF features are computed from smoothed input images. Smoothing reduces the impact of high-frequency changes that are often indicative of sensor noise or abnormal textures, which are known to negatively impact AdaBoost-based classifiers [9]. In this work we noticed that features produced from *smoothed* images are similar to *upscaled*, *unsmoothed* images. Therefore, we supposed that we could detect targets that are *smaller* than the template by upscaling the input image and adjusting the feature detection algorithm so that they do not pre-blur upscaled input imagery. To validate this, we trained a pedestrian detector to work with detection windows of 64x128 pixels using the INRIA pedestrian dataset [10]. We then ran several experiments where we degraded and then upscaled our test-image set by different scale factors from 1x all the way up to 10x. We computed an ROC curve for a full-resolution 64x128 classifier against the different test images and plotted them on a common plot. The goal was not to learn absolute performance on these test sets, but instead it was to ascertain how performance would degrade as we upscaled the test images more and more. Our results are shown in Figure 9. As expected, performance degrades the more we upscale the images, but what is surprising, perhaps, is that performance degrades very slowly from scale factors 1x to 4x, after which performance degrades rapidly. We exploit the fact that satisfactory performance can be achieves for small levels of upscaling. Features from images upscaled too much (>4x) are too degraded to be useful[2].

We configured our software to detect targets in detection windows as small as 16 pixels wide by 32 pixels tall (16 times smaller than the pedestrian detector in [8]) by building an image pyramid starting from an up-scaled version of the input image and by modifying our feature computation so that it omitted the blur step on upscaled images. A detection window of 32px tall corresponds to a target size of around 25px tall, after template padding. This statistic was validated in the experimental results (Section 4).

[2] This is not particularly surprising, as the local gradient histograms are computed on 4 pixel by 4 pixel windows.

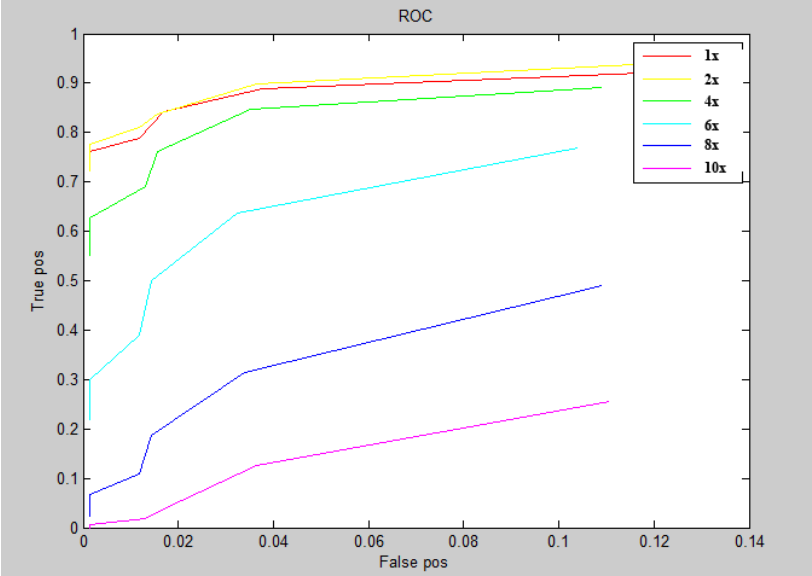

Figure 9. ROC curves for testing full-resolution classifier on different levels of upscaled test imagery.

Computationally efficient appearance based detection

The downside to blowing up the feature-pyramid is a large increase in computational complexity due to the quadrupling of the size of the base of the feature pyramid. We have found that maintaining a high processing rate in important for tracking performance as positional uncertainties can be kept small when the processing rate is fast. Simple upscaling the entire image by a factor of 4 leads to unsatisfactory tracking performance. In response to this, Toyon has developed a set of heuristics to reduce the computational complexity of our sliding-window detector when used in combination with our other detection and tracking approaches.

(1) Track Locality: First, we only run our sliding window classifier *near existing tracks*. The advantages of this are twofold – first, we decrease false-detection rates by taking advantage of a high *apriori* probability of a target being present in the region searched. For many scenes this means that large regions will never be searched – such as skyline, high trees, or occlusions – because there it is unlikely that a track is ever instantiated in that region. The second advantage is that the feature pyramid only needs to be computed for each region near tracks. In our processing flow, we compute the 3-sigma uncertainty ellipse and then compute the minimum enclosing rectangle for that ellipse. Next, we merge rectangles that intersect one another. Finally, we pad the region in such a way to account for the size of our detection window, which is computed using the ground-plane and the classifier ground sample distance. For typical scenes, the searched area is dramatically reduced from 4 times the *original* image size down to on the order of 4 times the *target* size. This approach does have the disadvantage of being unable to detect stationary targets; however, for a large number of applications a target is only interesting if it is moving for at least some amount of time. Therefore, if we have a reliable motion detector, we can simply wait until the target moves before we start trying to classify it. We have found this approach to be more robust than simply doing a global-search for a given target class.

(2) Temporal Preference: The above optimization dramatically reduces the image search window but can still run into computational bottlenecks in certain difficult circumstances involving scenes with many targets. For scenes when the following conditions are met, computation throughput decreases to unacceptable levels: (1) there are tracks on several targets, (2) the targets are widely separated, (3) there is a large position-uncertainty, and (4) the geometry is such that detection of those targets requires computation of a blown-up feature pyramid – this happens in geometries where the camera field-of-view is nearly parallel with the ground-plane. In cases where these conditions are met we have found it beneficial to *omit* areas of the image from the sliding-window search in order to allow for the collection of detection algorithms to process at full frame-rate. To implement this, we put a hard-limit on computation per frame. In any given image we choose a maximum search area size (e.g. 640x480) and for frames where search area is larger than the maximum search area we will randomly sample a

sub-region to be searched. This maintains high frame-rates overall (while the sliding-window does not necessarily cover the entire search area), which allows motion-detections and feature-detections to reduce target positional uncertainty. This improves tracking performance overall, but also alleviates one of the factors that leads the computational bottleneck in the first place, which was high positional uncertainty[3].

(3) Scale Consistency: The previous optimizations address the computation of the feature-pyramid. Our third optimization reduces the complexity of the sliding window classifier by using the ground-plane model and our object-models to deduce the approximate target size at every location in the image. We know that humans will not be very large at distant points and that humans will not be too small at nearby points, and our ground-plane model tells us all that we need to generate a *mask* that informs a sliding-window classifier where to search for a given pyramid level. For a planar model, the set of locations to evaluate will be the set of points at a particular range, covering line in the image. Absent this mask, we would have to search the entire image. This amounts to over an order-of-magnitude reduction in the number of areas the sliding-window classifier must evaluate at a given scale. In Figure 10 we show an example of search regions for scales of the sliding-window classifier. Note that the search region has some "thickness" to it to allow for a small (~20%) margin of error in the target size estimate.

The combination of search strategies above leads to a dramatic reduction in the total number search locations. Indeed, the reduction is more than enough to overcome the increase in computational requirements caused by the image-pyramid expansion discussed in the previous section.

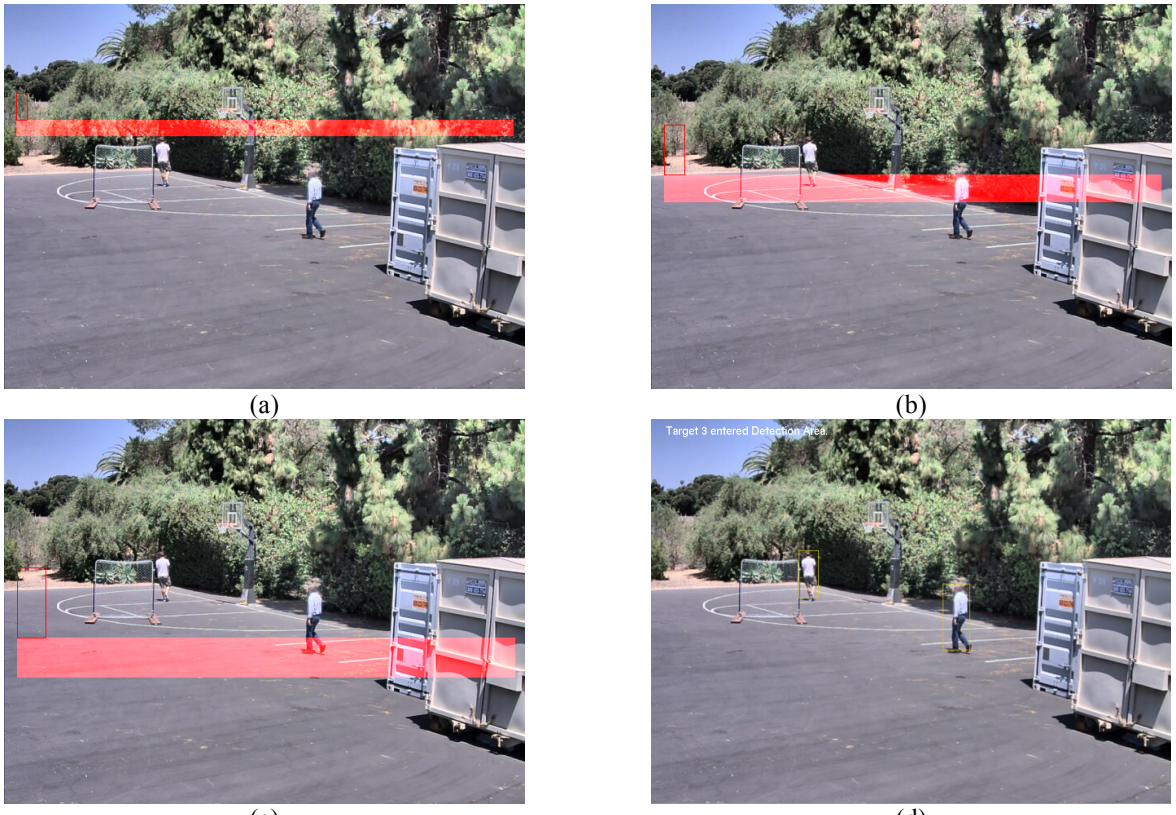

Figure 10. Sliding window mask at different scales. In (a)-(c) we show the sliding window mask for different pyramid levels. A candidate rectangle is shown in red. In (d) we show detection of both dismounts at the correct size.

[3] This is another example of how the different detection methods complement one another.

Tracking

To provide increased probability of detection and reduced false alarm rate in high-clutter environments, we employ tracking algorithms to filter individual target detections and fuse them onto target *tracks*. After camera motion compensation and change detection processing, individual pixel change detections are processed to identify clusters of changed pixels corresponding to potential moving objects (a cluster may consist of a single or multiple pixels). Moving target measurements are then generated in the image plane as the center of mass of each cluster. Each measurement is then (a) associated and fused with an existing target track, (b) used to initiate a new target track, or (c) declared a *false detection* and discarded. The central challenge in multi-target tracking is the data-association problem, which involves deciding between options (a), (b), or (c) for each new detection. To solve the data-association problem we have employed the Joint Probabilistic Data Association Filter (JPDA) [11]. This filter considers the probability of several joint associations of detections to tracks and clutter. By considering associations jointly the algorithm precludes many problems associated with greedy assignment approaches. After probabilistically associated detections with tracks and/or clutter, tracks are updated using a modified Kalman filter update rule.

Because track estimates are based on the entire history of detections, high probability of detection is obtained, even for slowly moving targets, and since cues are generated only for tracks, which are highly likely to indicate the true location of a moving target, the false alarm rate is greatly reduced. Since video sensors provide imagery at a high frame rate, tracks are confirmed quickly, typically in less than one second. In Figure 11 we show tracks of four targets that are persistently tracked. All previously discussed detection methods are integrated together by the tracker.

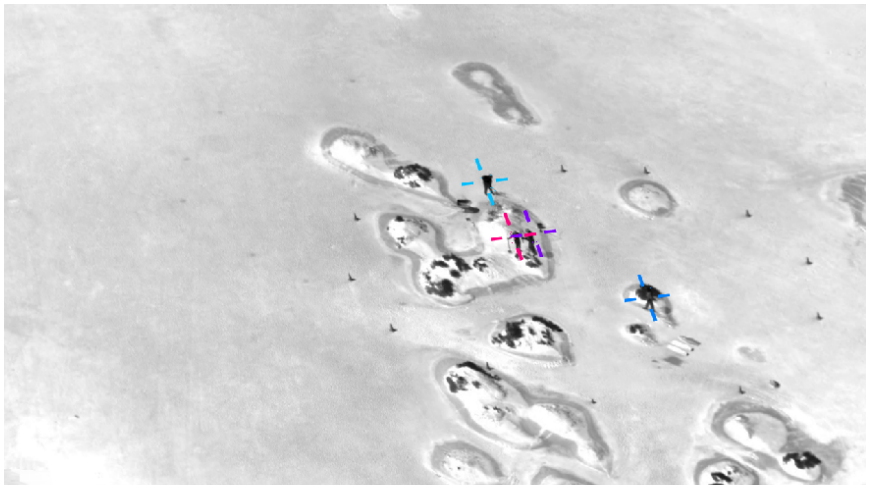

Figure 11: Persistent tracking of multiple closely-spaced targets in MWIR video captured from a Robinson R44 helicopter.

Target class and size information is estimated using measurements that associated with tracks high probability (>99%). Target class is fused using simple Bayesian fusion. Activities are recognized as by detecting tracks of a particular class in a user-defined keep-out area. In experiments discussed in Section 3, for instance, we generate alerts for detected humans located in a region between two fences.

3. EXPERIMENTAL SETUP AND TESTING

An evaluation of the VideoPlus®-Aware system was performed at the Y-12 National Security Complex and McGhee Tyson Air Base in partnership with the Consolidated Nuclear Security, LLC. The Performance Test was designed to evaluate the ability of the VPA system to perform autonomous video-based moving target detection, tracking and classification. Additionally, the test was also configured to evaluate the ability of the VPA system to classify basic actions such as the breaching a fence-line for entry into a secure area.

Prior to the start of the Performance Test, a Test Plan was drafted by Y-12 and TOYON. The Test Plan described the primary evaluations, the evaluation metrics, and the expected minimum levels or performance that was expected from the system. Testing was divided into three categories: (1) dismount detection and classification at short range, (2) dismount detection and classification at long range, and (2) classification of actions of interest. Primary metrics of

interest were: probability of correct detection, probability of correct classification (of dismounts and actions), false alarm rate (FAR), and nuisance alarm rate (NAR). These metrics are explained in more detail in the Metrics section.

Test locations

The cameras were on the roof of the TTF at a height of 9.1m (30') above the ground. This height combined with the natural contours of the test facility and the focal length of the visual camera provided the field of views (FOVs) shown in Table 1 (Thermal was slightly smaller).

The test facility includes a simulated double row Perimeter Intrusion Detection and Assessment System (PIDAS) fence located at 27.1m (89') and 33.2m (109') with a crushed limestone gravel bed between fences and extending to 10' on either side, as seen in Figure 3. The fences were 8' chain link with three strands of bare wire on top as simulated barbwire. The ground has a gradual slope away from the cameras to approximately 43m and then drops away quickly to a tree line 70m away.

Table 1. FOV at TTF

Distance	**FOV width**	**Target size**
20m (65.6')	9.8m (32')	142 pixels
25m (82')	12.2m (40')	114 pixels
30m (98.4')	14m (46')	95 pixels
40m (131')	18.3m (60')	71 pixels
50m (164')	23.7m (78')*	57 pixels
60m (196.9')	28.5m (93.5')*	47 pixels
70m (229.7')	33.3m (109')*	41 pixels
* Due to FOV obstructions, only ~1/2 the FOV was useable		

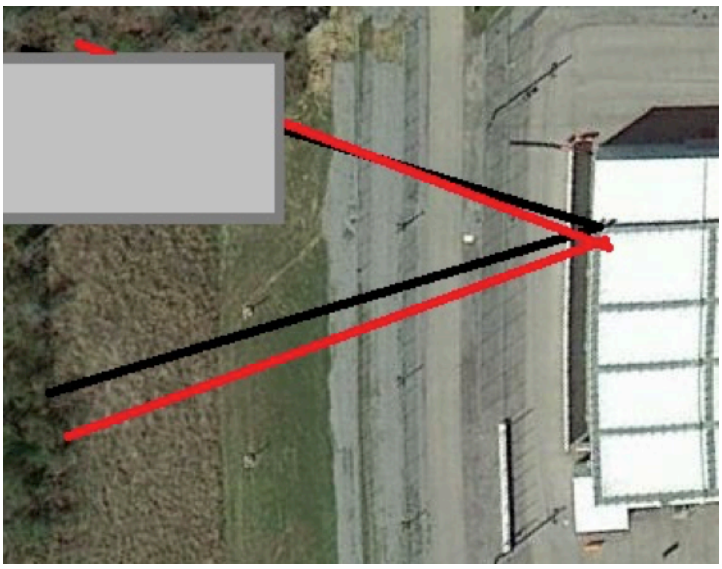

Figure 12: TTF fields of view. The red lines indicate the visual camera FOV and the black lines indicate the thermal camera FOV. The gray box indicates unusable portions of the respective FOVs.

The second location is the McGhee Tyson Air Base (MTAB) located in Alcoa, TN. MTAB provided a secure area for longer range testing. This was the location of the tests described in paragraph 5.3. Two different sets of cameras (visual and thermal) were set up on a hillside at an elevation of 304m (997'), pointing in different directions, as shown in Figure 13. FOV 1 included visual and thermal cameras capturing a wide field of view; consequently, targets in FOV 1 had very poor resolution, and appeared very small within the senor frame. FOV 2 included visual and thermal cameras capturing a narrow field of view; consequently, targets in FOV 2 had better resolution than FOV 1, and appeared larger within the

senor frame. The camera mounting positions combined with the natural contours of the test facility and the focal length of the visual cameras provided the FOVs shown in Table 2 (Thermal was slightly smaller).

Table 2. FOV at MTAB

Distance	FOV width	Target Size
FOV 1		
250m (820')	33.5m (110')	46 pixels
500m (1640.5')	67.1m (220')	23 pixels
FOV 2		
500m (1640.5')	9m (30')	115 pixels
700m (2296.5')	25.3m (83')	82 pixels

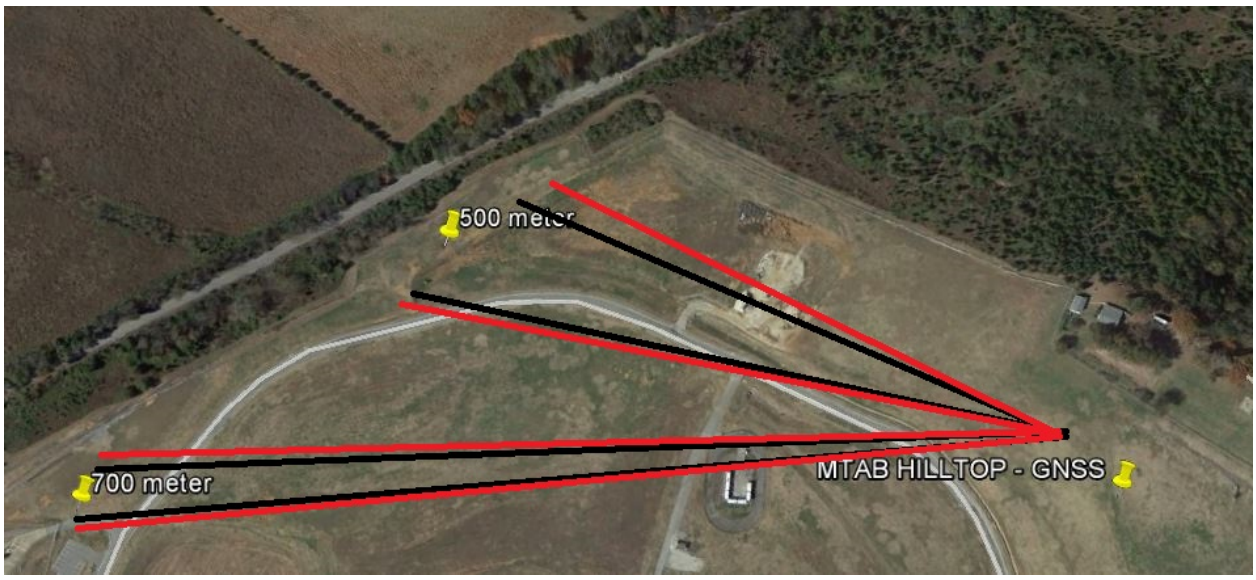

Figure 13: MTAB fields of view for tests at 700m and 500m. The red lines indicate the visual camera FOV and the black lines indicate the thermal camera FOV. Note that the test at 500m covers a significantly larger width.

The ground elevation dropped off quickly from the camera location but then rose with a gentle incline from 170m range (to camera) out to 475m range in FOV1. Similarly, the ground elevation rose with a gentle incline from 240m range out to 650m range in FOV2. FOV1 included a storm-water containment berm at 500m and FOV2 included a small hilltop at 700m.

Testing was conducted during four contiguous days on June 16, 2014, through June 19, 2014. All testing was performed during dry weather with no participation or condensation, present. Daily temperature ranges during the test period were 19.5 to 33C (67 to 91F), daily Relative Humidity ranged from 55 to 89%. Visibility was unlimited.

Cameras

Both Visual (or Visible) spectrum and Thermal Infrared spectrum cameras were used for the test.

Visual spectrum cameras are defined as electro-optical imagers with the ability to detect radiation generally within the electromagnetic spectrum visible to humans, with the addition of some sensitivity in the near infrared and near ultraviolet ranges. This includes wavelengths of 300 to 700 nanometers.

Thermal cameras are defined as electro-optical imagers with the ability to detect within the medium and long wavelength infrared spectrum. This includes the wavelengths of 3,000 to 13,000 nanometers.

Cameras used for the TTF testing

Thermal camera used for short range testing at the TTF

IEC Infrared Corporation M1-NSTI-D25-IF0-W Imaging System

Array Format	640x480 pixels
Detector Type	Vanadium Oxide (VOx)
Cooling	Uncooled
Spectral Range	8-14 micrometers
NETD	< 50 millikelvin
Imaging Rate	60 Hz
Cold Start Time	Less than 30 seconds @ 23°C
Lens	24.4.o-2.8o Continuous Zoom F1.5 Germanium optics

Visual camera used for short range testing at the TTF:

Axis Q1765-LE

Image sensor	1/2.9" progressive scan RGB CMOS
Array Format	1920x1080 pixels (max), 640x480 pixels (as used)
Spectral Range	300-700 nanometers. (Visual Range)
Imaging rate	30 Hz
Lens	f=4.7–84.6 mm, F1.6–2.8

Cameras used for the MTAB Testing

Thermal camera used for long range testing at MTAB (700m):

IEC Infrared Corporation M1-NSTI-M39-IF0-W Imaging System

Array Format	640x512 pixels
Detector Type	Indium Antimonide (InSb)
Cooling	Stirling cooled
Spectral Range	3-5 micrometers
NETD	< 25 millikelvin
Imaging Rate	60 Hz
Lens	48mm-701mm /11.4-0.8 F-5.5 Germanium optics

Visual camera used for long range testing at MTAB (700m):

IEC Infrared Corporation

Lens	23mm-506mm/ 15.8-0.7
Array Format	630,000 pixels
Spectral Range	300-750 nanometers. (Visual Range)

Thermal camera used for medium range testing at MTAB (250m-500m):

FLIR PT-series

Array Format	640x512 pixels
Detector Type	FPA, uncooled Vanadium Oxide (Vox) microbolometer
Cooling	uncooled
Spectral Range	7.5 to 13 micrometers
Thermal sensitivity	<50mK f/1.0
Imaging Rate	25 Hz
Lens	Focus free, thermal lens

Visual camera used for medium range testing at MTAB (250m-500m):

FLIR PT-series

Image sensor	1/4” Exview HAD CCD
Array Format	640x480 pixels (as used)
Spectral Range	300-700 nanometers. (Visual Range)
Imaging rate	25 Hz
Lens	f=3.4mm (wide) to 122.4 mm (tele), F1.6 to F4.5

Data collection and processing

Data collection occurred before, during and after the actual tests were performed. Video streams from the VPA system included all markups such as detection and classification. The review of these streams provide the basis for Detection rates and FAR and NAR. During testing, Y-12 test observers collected the necessary test data using custom software (Figure 14) running on tablet computers. This data included test start and stop times, detection, object classification and action classification. All testing and associated detection, tracking and classification messaging was confirmed during testing by observation of the data reported to RaptorX. All RaptorX logs were saved for evaluation after testing.

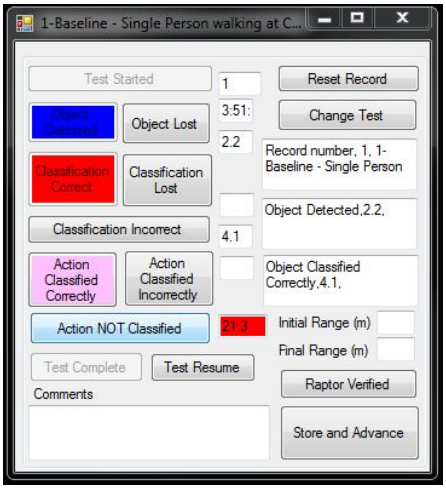

Figure 14. Data Collection Software

Tests performed

Dismount detection and classification

The primary focus of the first tests was to evaluate the detection (sensing), tracking, and classification of objects in the sensor field of view (FOV). Test subjects included single and multiple human beings, canines, and canines with humans. Additional items of interest occasionally appeared in the scene, including birds, insects (flying very close to the camera), a tractor, an elk, and blowing foliage.

For the purpose of our testing, all moving objects were considered to be either threats (dismounts) or non-threats (non-human objects). The purpose of the test was to determine the software's ability to sense and classify all upright human subjects as Threats. Tests included the subjects walking perpendicular to the camera's FOV and subjects walking directly at the camera.

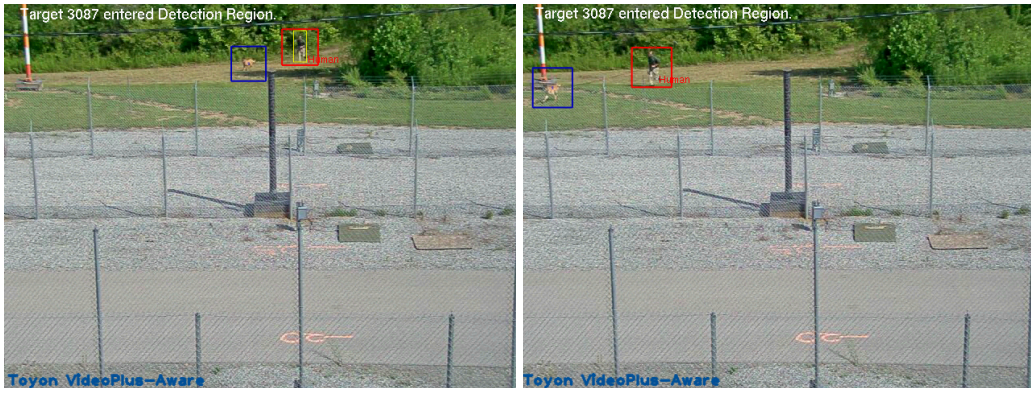

Figure 15. A canine and a human are detected, tracked, and correctly classified.

Test Performed:

Humans

1-Baseline - Single Person walking at the Camera

2-Baseline - Multiple people, spaced 15 seconds apart, walking at the Camera

3a-Perpendicular - Single person walking across the Camera FOV at preset distances

3b-Perpendicular - Multiple people walking across the Camera FOV at preset distances

Canines

4a-Canine at the Camera (Canine alone and Canine with Handler)

4b-Canine across the Camera FOV at preset distances (Canine alone and Canine with Handler)

Long-range detection and classification

The primary focus of these tests was to evaluate the detection (sensing), tracking, and classification of objects at longer ranges. Subjects included single and multiple human beings, canines and canines with humans. Additional items of interest often appeared in the scene, including pedestrians, bicycles, cars, trucks, and tractor-trailers. Tests included the subjects walking perpendicular to the camera's FOV and subjects walking directly at the camera.

Test Performed

Humans

1-Baseline - five people walking at the Camera spaced 30 seconds apart

2-Perpendicular - Multiple people walking across the Camera's FOV at preset distances

Canines

3a- Canine at the Camera (Canine alone and Canine with Handler)

3b-Canine across the Camera's FOV at preset distances (Canine alone and Canine with Handler)

Figure 16. Long Range Detection at ~700m range from sensor to target.

Classification of action

The primary focus of these tests was to evaluate the classification of specific actions by the objects. Subjects included single and multiple human beings. For the purpose of our testing, the action that the security system was attempting to classify was breaching a fence-line. Tests included subjects walking perpendicular to the camera's FOV and breaching a

fence and subjects walking directly at the camera and breaching a fence. Additional items of interest occasionally appeared in the scene, including birds, insects (flying very close to the camera), a tractor, an elk, and blowing foliage.

Test Performed:

Humans

1-Baseline - Single person walking at the Camera and Breaching Fence

2-Baseline - Multiple people, spaced 15 seconds apart, walking at the Camera and Breaching Fence

3a-Perpendicular - Single Person walking down the fence line and Breaching Fence

3b-Perpendicular - Multiple people walking down the fence line and Breaching Fence

Metrics

The following metrics were defined and used to evaluate the performance of the Toyon VPA system.[4]

Probability of Sensing (or Initiation): A *Sensing* or *Initiation* event occurs when an object in the sensor FOV is automatically detected by the security system. The *Probability of Sensing (Ps)* (or *Probability of Initiation*) is defined as the ratio of Sensing events to targets appearing in the sensor field of view; in other words, *Ps* is the probability that the security system will sense a target within its FOV and begin a target Initiation. The VPA system indicates a sensed object by a blue box surrounding the target. The Sensing event is based on motion or other changes in the FOV, and does not imply that the target has been classified.

Probability of Assessment (or Classification): An *Assessment* or *Classification* event occurs when a Sensed human target in the sensor FOV is automatically classified as a Threat by the security system. For the purpose of this test and report, any **human** in the FOV was considered to be a threat (which is a reasonable assumption for surveillance of a secure perimeter). *Probability of Assessment (Pa)* (or *Probability of Classification*) is defined as the ratio of Assessment events to targets; in other words, *Pa* is the probability that the security system will correctly classify a threat. (This metric is commonly called ***Probability of Correct Classification*** or *Pcc* in the video processing literature.)

Probability of Correct Classification of Action: A *Correct Classification of Action* occurs when a Sensed object in the sensor FOV is performing an action, and the action is correctly classified by the security system. *Probability of Correct Classification of Action (Pca)* is the ratio of Correct Classification of Action events to Sensed human targets in the FOV performing the action. In this test, the classification of actions was limited to *humans breaching a secure area*.

Probability of Detection: A *Detection* event occurs when an object in the sensor FOV has been Sensed and Assessed as a Threat. A Detection event triggers an alert in the security system. The *Probability of Detection (Pd)* is the ratio of Assessment events to targets in the FOV. *Pd* is the product of the probability of sensing and the probability of assessment (*Pd = Ps * Pa*).

Probability of Detection of Action: A *Detection of Action* event occurs when an object in the sensor FOV is performing an action, and the action is correctly classified by the security system. A Detection of Action event triggers an alert in the security system. The *Probability of Detection of Action (Pda)* is the ratio of Correct Classification of Action events to targets in the FOV objects performing the action. *Pda* is the product of the probability of sensing and the probability of assessment of a given sensor (*Pda = Ps * Pca*).

False Alarm Rate (FAR): For the purposes of this evaluation, a *false alarm* was an alarm generated within the system when there is no observable stimulus presented to the sensor, and the cause of the alarm cannot be determined. False alarms can be an indication of electronic malfunction or failure, including communications and power systems. The occurrence of False Alarms is meaningful because it may affect operator confidence in the system. The number of false alarms divided by total system runtime indicates the *False Alarm Rate (FAR).*

Nuisance Alarm Rate (NAR): A *nuisance alarm* is an alarm generated by a stimulus that the system was designed to detect, but produced by a source other than an intruder. This could include alarms on animals, wind-blown tree branches,

[4] The reader should note that these definitions differ from the standard definitions often used in video processing literature. These definitions were chosen by Y-12 to maintain consistency with evaluations of similar DOE security technologies.

or rapid changes in lighting or cloud cover. The number of nuisance alarms divided by total system runtime indicates the *Nuisance Alarm Rate (NAR)*.

4. RESULTS

The following sections summarize results from the four-day performance test conducted by the Y-12 National Security Complex in Oak Ridge, TN. The results presented in this paper are aggregate results from all of the individual tests performed over the four-day evaluation. Results from the individual tests were left out of this paper for brevity, but they do appear in the Y-12 Test Report [12].

Object detection and classification

During testing, data collection captured 811 test data points. Of these, the combined thermal and visual systems detected an item of interest 810 times for a Ps of 99.9%. The thermal system collected 494 of the data points with a Ps of 100% while the visual camera was responsible for the remaining 317 data points and a Ps of 99.7%.

Of the 811 data points, 95 were collected with no expectation of classification. This was due to the test subject being a canine. The data shows a Ps of 100% for the canines with no false classifications.

In the remaining 716 tests, the test subject was a walking human and there was always the expectation of classification. Of these tests, 715 initiations occurred for a Ps of 99.9%. The consistency of the initiations was high and very reliable with a few exceptions. Correct classification as human occurred in 617 data points for a Pa of 86.2%. The Pa was above the Test Plan's Threshold Pa/Pcc of 50% and the Objective Pa/Pcc of 80%. The observed Ps and Pa provided a Pd of 86.1%. The thermal system collected 443 of the data points with a Ps of 100% and 383 classifications for a Pa of 86.5% resulting in a Pd of 86.5%. The visual camera was responsible for the remaining 273 data points and a Ps of 99.6% and 234 classifications for a Pa of 85.7% resulting in a Pd of 85.4%.

Table 3. Classification performance

Comparison of Results with Expected Performance (as defined in the Test Plan)					
Feature	**Imager**	**Metric**	**Test Plan: Threshold Performance**	**Test Plan: Objective Performance**	**Achieved Performance**
Sensing (Initiation)	Visible	Ps	> 0.80	> 0.90	**.996**
	Thermal	Ps	> 0.80	> 0.90	**1.00**
Assessment (Classification)	Visible	Pcc / Pa	> 0.50	> 0.80	**.857**
	Thermal	Pcc / Pa	> 0.50	> 0.80	**.865**
Detection (Ps * Pa)	Visible	Pd	Not provided	Not provided	**.854**
	Thermal	Pd	Not provided	Not provided	**.865**

Long range detection and classification

Testing included 329 data points for evaluating detection and classification of test subject at longer ranges. During these tests, 329 initiations occurred for a Ps of 100% with 270 human classifications for a Pa of 82.1%. This was above the Threshold Ps of 80% and Objective Ps of 90%. This data provides a Pd of 82.1%. The thermal system collected 233 of the data points with a Ps of 100% and 199 classifications for a Pa of 85.4% resulting in a Pd of 85.4%. The visual camera was responsible for the remaining 96 data points with a Ps of 100% and 71 classifications for a Pa of 74% resulting in a Pd of 74%. All of the data points with no classification occurred during tests at 500m or walking from 500 to 250m; these tests provided the lowest target resolutions (fewest pixels on target).

Table 4. Long Range Classification, Expectations vs. Data

Comparison of Results with Expected Performance (as defined in the Test Plan)					
Feature	**Imager**	**Metric**	**Test Plan: Threshold Performance**	**Test Plan: Objective Performance**	**Achieved Performance**
Sensing (Initiation)	Visible	Ps	> 0.80	> 0.90	**1.0**
	Thermal	Ps	> 0.80	> 0.90	**1.0**
Assessment (Classification)	Visible	Pcc / Pa	Not provided	Not provided	**.74**
	Thermal	Pcc / Pa	Not provided	Not provided	**.854**
Detection (Ps * Pa)	Visible	Pd	Not provided	Not provided	**.74**
	Thermal	Pd	Not provided	Not provided	**.854**

Action classification

Testing included 200 data points with the expectation of correctly classifying the test subject breaching the secure area. The system must first detect the object and then classify it as human. If the classified object is determined to be in the secure area, then the system classifies its actions as a **breach alarm**. During these tests, 200 initiations occurred for a Ps of 100%, with 186 human classifications for a Pa of 93% and 164 classified breaches for a Pca of 82%. These were above the Threshold marks of 50% and the Objective marks of 80%. The data provides a Pd of 93% and a Pda of 82%. The thermal system collected 120 of the data points with 120 initiations for a Ps of 100%, 116 human classifications for a Pa of 96.7% and 116 action classifications for a Pca of 96.7%. This resulted in a Pd and a Pda of 96.7%. The visual camera was responsible for the remaining 80 data points with 80 initiations for a Ps of 100%, 70 human classifications for a Pa of 87.5% and 48 action classifications for a Pca of 60%. This resulted in a Pd of 87.5 and a Pda of 60%.

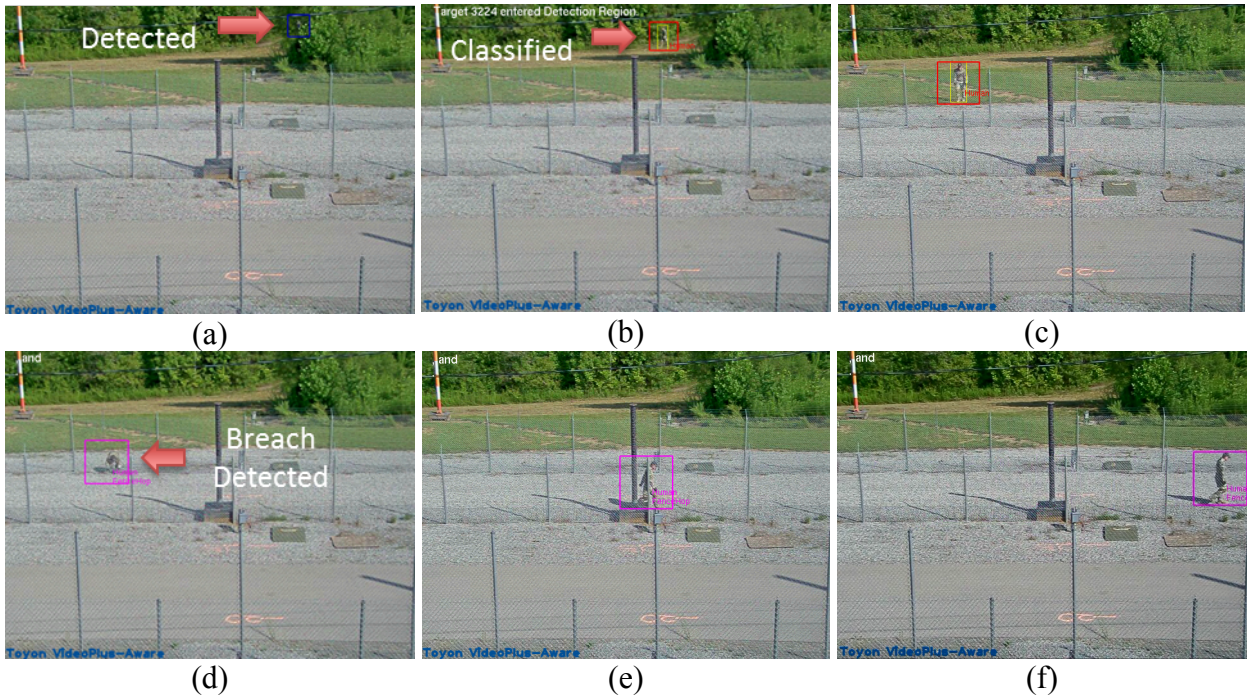

Figure 17: VideoPlus®-Aware (a) Detects and tracks a target (blue box) in the distance and in (b) classifies it as a human (red box). (c) the target is continuously tracked as it approaches the line of detection. (d) A fence-breach is detected (purple box) as the target crawls through a hole in the fence. In (e) and (f) the suspect is tracked as he exits the field of view. This video was recorded automatically by VPA in response to the action.

Table 5. Action Classification, Expectations vs. Data

Comparison of Results with Expected Performance (as defined in the Test Plan)					
Feature	**Imager**	**Metric**	**Test Plan: Threshold Performance**	**Test Plan: Objective Performance**	**Achieved Performance**
Sensing (Initiation)	Visible	Ps	> 0.8	> 0.9	**1.0**
	Thermal	Ps	> 0.8	> 0.9	**1.0**
Assessment (Classification of Action)	Visible	Pca	> 0.5	> 0.8	**.60**
	Thermal	Pca	> 0.5	> 0.8	**.967**
Detection of Action (Pda = Ps * Pca)	Visible	Pda	Not provided	Not provided	**.60**
	Thermal	Pda	Not provided	Not provided	**.967**

FAR and NAR

Video review of system runtime provided the data for FAR and NAR. These numbers, added to the observed data from the organized test cycles, provided the overall picture of system performance. Runtime during system setup and training was ignored.

FAR

The evaluations include 30.5 hours of runtime for the system. This runtime included the 811 data points and operational time before, during and after the testing cycles. This operational time observed no false alarms, providing a FAR of 0 per hour.

NAR

The operational time observed 22 verifiable nuisance alarms, providing a NAR of .72 per hour. Seventeen alarms involved the visual camera and moving vehicles, with 15 of the 17 during one 1.75-hour span of long range testing. Only one alarm occurred involving the thermal camera. Four alarms involved the visual camera and a preset, vertical 6" by 96" pipe.

Table 6. FAR/NAR, Expectations vs. Data

Comparison of Results with Expected Performance (as defined in the Test Plan)					
Feature	**Imager**	**Metric**	**Test Plan: Threshold Performance**	**Test Plan: Objective Performance**	**Achieved Performance**
Detection (Ps * Pa)	Visible	FAR	< 3/hr	< 2/hr	**0/hr**
		NAR	< 3/hr	< 2/hr	**.69/hr**
	Thermal	FAR	< 3/hr	< 2/hr	**0/hr**
		NAR	< 3/hr	< 2/hr	**.03/hr**

5. CONCLUSIONS

The research presented in this paper includes the development, testing, and evaluation of algorithms and software that were specially designed to automatically detect, track, and classify dismounted targets in video, while maintaining very low false and nuisance alarm rates on non-human targets. These algorithms and software were designed as an additional feature to Toyon's existing VideoPlus tracking product. These algorithms were developed and extensively tested in the office/lab environment, before being evaluated in a thorough multi-day performance test conducted by the government. The performance test included several different visible and thermal cameras, two different test environments, short-range

and long-range tests, and "confuser" targets (including vehicles, canines, and elk) designed to fool the VPA sysem and generate nuisance alarms. Overall results were very good, showing that the system could achieve high rates of dismount detection and classification while maintaining low FAR and NAR rates. This technology was also packed into a low-SWaP computing platform for easy deployment in many different environments.

Future research will continue to improve the robustness of the algorithms to environmental variations such as rapid changes in lighting and temperature, snow, rain, and marine environments. We are also developing an algorithm that will be more robust to detecting dismounts that are partially obscured, kneeling, or crawling. We also have plans to develop multi-sensor data fusion logic which would correlate dismounted targets from multiple sensors in order to "stich" the target track across multiple sensor FOVs.

ACKNOWLEDGEMENTS

Toyon would like to thank Joe Rainwater, Garret Scott, and Andrew Sharp of Consolidated Nuclear Security for substantial contributions to the performance evaluation of VideoPlus®-Aware. Toyon would also like to thank the following contributors for their generous support of this research:

- The Air Force Research Lab (AFRL/RWK) under SBIR contracts FA8651-11-M-0087 and FA8651-12-C-0081.
- The Air Force SBIR Commercialization Readiness Program (AFRL/XPPD) under SBIR contract FA8651-12-C-0081
- Mind's Eye Program; DARPA Transform. Conv. Tech. Off. (TCTO), Contract No.: W911NF-10-C-0084.

REFERENCES

[1] R. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: a systematic survey," *IEEE Transactions on. Image Processing*, v. 14, no. 3, March, 2005.
[2] C. Stauffer and W. E. L. Grimson , "Adaptive mixture models for real time tracking," *Proc. CVPR,* 1999.
[3] A. P. Brown, K. J. Sullivan, and D. J. Miller, "Feature-aided Multiple Target Tracking in the Image Plane," Proceedings of the SPIE Conference on Intelligent Computing: Theory and Applications IV (Orlando, FL), vol. 6229, April 2006.
[4] R. Benenson, M. Omran, J. Hosang, B Schiele, "Ten Years of Pedestrian Detection, What Have We Learned?" ECCV, 2014.
[5] 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.
[6] Jianbo Shi and Carlo Tomasi. Good Features to Track. IEEE Conference on Computer Vision and Pattern Recognition, pages 593–600, 1994.
[7] Martin A. Fischler and Robert C. Bolles (June 1981). "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". Comm. of the ACM 24 (6): 381–395.
[8] Dollár, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. PAMI (2014).
[9] Freund, Shapire, "A short introduction to Boosting," Journal of Japanese Society for Artificial Intelligence, September 1999.
[10] Dalal, N. and Triggs, B. and Rhone-Alps, I. and Montbonnot, F., "Histograms of oriented gradients for human detection," *IEEE Conference on Computer Vision and Pattern Recognition*, 2005.
[11] Yaakov Bar-Shalom and Thomas E. Fortmann, "Tracking and Data Association," Academic Press, 1988.
[12] J. Rainwater, G. Scott, A. Sharp, "Test Report for the Evaluation of VideoPlus-Aware," Consolidated Nuclear Security, LLC., July 2014.