

ARMY RESEARCH LABORATORY



**Force Protection Surveillance System:
Algorithm and Performance**

by Alex Lipchen Chan

ARL-TR-5322

September 2010

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Adelphi, MD 20783-1197

ARL-TR-5322

September 2010

Force Protection Surveillance System: Algorithm and Performance

Alex Lipchen Chan
Sensors and Electron Devices Directorate, ARL

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188		
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) September 2010		2. REPORT TYPE		3. DATES COVERED (From - To) January 2004 to July 2010	
4. TITLE AND SUBTITLE Force Protection Surveillance System: Algorithm and Performance			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Alex Lipchen Chan			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory ATTN: RDRL-SES-E 2800 Powder Mill Road Adelphi, MD 20783-1197			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-5322		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The Force Protection Surveillance System (FPSS) was developed to address the increasing needs of physical security at critical facilities. Central to the FPSS is a robust target tracking algorithm that consists of background modeling, target detection, and target tracking modules. In this report, a detailed algorithmic description is provided for each of these modules, which is supplemented with the corresponding flowcharts in the appendix. An initial assessment on the performance of FPSS was carried out using a large and realistic dataset that contains 53 pairs of concurrent color-infrared image sequences. The results of this assessment are analyzed and presented in this report, which show excellent detection rates and acceptable false-alarm rates for this challenging dataset.					
15. SUBJECT TERMS Force protection, target detection, target tracking, background modeling					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (Include area code)
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UU	36	Alex Lipchen Chan (301) 394-1677

Contents

List of Figures	iv
1. Introduction	1
2. FPSS Algorithm	4
3. Experimental Results	17
4. Conclusions	24
5. References	25
Appendix A. Flowcharts for the FPSS Algorithm	29
List of Symbols, Abbreviations and Acronyms	33
Distribution List	34

List of Figures

Figure 1. Left image shows a number of video surveillance cameras installed on a street light pole. Right image shows a surveillance control room with a wall of video surveillance monitors.	2
Figure 2. The schematic diagram of the background modeling process in FPSS.....	6
Figure 3. The effects of stability mask on background models. Left image shows that active foreground pixels (identified with red ovals) could seep in and ruin the background models when the stability mask is deactivated. Right image shows a much cleaner background model when the stability mask is activated.....	8
Figure 4. Enhancement of target signatures and suppression of trailing effects and noises are achieved simultaneously in a DPI. Two difference images on the left show the trailing effects and random noises in different places, which are suppressed in the resulting DPI. Signatures of legitimate moving targets appear in the same locations on both difference images, thus, they are enhanced in the DPI through the multiplicative process.....	9
Figure 5. Polarity changes are common in FLIR cameras equipped with automatic gain control. Left image shows that human signatures are clearly brighter than their background in a typical winter afternoon. Right image shows that human signatures are much darker than their background in the same parking lot during summer time.	10
Figure 6. The upper-left image is an image frame in a FLIR video sequence. The upper-right image is the corresponding DPI with some small but clearly visible noises. These noises are removed after the morphological operation (lower-left). The “blobiness” of moving targets is increased in the lower-right image through a pyramid-means method.	11
Figure 7. Three moving targets are detected on this input frame, which are tagged as Targets 1, 2, and 3, respectively. The size of each target is delimited by a rectangular bounding box and the potential type of target is represented by the color of the box. The green area is a “don’t care” zone, while the red area is a “critical” zone.....	12
Figure 8. FPSS change detection capability. A man left a suitcase on the traffic island adjacent to parking lot (lower-right and middle images). FPSS detected the scene change and highlighted the abandoned object within a few seconds (upper-left image).....	13
Figure 9. The FPSS graphical user interface.	14
Figure 10. Using the track history (purple arrow) of a moving target, the FPSS predicts the likely directions and locations (yellow arrows) of the target in the near future.	16
Figure 11. Analyzing the track history (red arrows) is useful in establishing a normal pattern of movements and detecting suspicious movements in a scene.	17
Figure 12. The Sentry POD used in the FPSS data collection effort (left) and a typical view of a parking lot (right).....	19
Figure 13. A pair of concurrent Color-FLIR images in the Second FPSS Dataset that was coarsely co-registered.	19

Figure 14. The graphical user interface used for the FPSS ground-truthing effort.20
Figure 15. The ROC curve for the color (top) and FLIR (bottom) sequences, respectively.22

INTENTIONALLY LEFT BLANK.

1. Introduction

As the threats from terrorists and criminals appear to be intensifying dramatically over the last decade, both the public and private sectors have invested an enormous amount of resources to address concerns of physical and personnel security. Due to the highly informative and easily understood nature of video display, video surveillance has long been embraced by security personnel in the form of closed-circuit television (CCTV). Significant improvements in sensor technology, digital communication, data storage, and computing power have enabled and empowered the exponential growth of video security surveillance in recent years. At a reasonable cost, practically any organization can install dozens of high-resolution, low-lux, and pan-tilt-zoom video security cameras around its facility, display the video outputs on a wall of large and bright liquid crystal display (LCD) monitors, and store the video data into digital video recorders (DVRs) with terabytes of memory. In their 2006 report, Ball et al. (1) estimated that there might be as many as 4.2 million CCTV cameras in Britain, and a person could be captured on over 300 cameras each day. The left image in figure 1 shows a group of surveillance cameras installed on a street light pole, which could have contributed to these surveillance statistics in Britain and elsewhere.

Unfortunately, most of the surveillance video data in the world were collected, displayed, stored, overwritten, and forgotten over time, without anyone ever having promptly and reliably extracted the vital actionable information that could have been embedded in the ever-increasing streams of data. This happens because human operators are simply unable and unwilling to monitor so many video streams over a long period of time. In their studies, Sears and Pylyshyn (2) showed that a typical human operator may track up to four moving targets simultaneously, but is unable to keep up with neighboring distracters efficiently. Both the number and speed of moving targets in the video streams are inversely related to the monitoring performance of human operators (3). Furthermore, the tracking capabilities of human operators are obviously bounded by spatial and temporal limits; thus, the effectiveness of monitoring degrades quickly as the wall of monitors becomes larger and the duration of screen-watching stretches longer. In many force protection scenarios, the video streams carry either near-static or rather mundane views most of the time. As a result, human operators often lower their vigilance level and rarely watch any video surveillance stream at all. The right image of figure 1 shows the reality inside the control room of a typical video surveillance network. Most of the time, one or none of the many video surveillance cameras is actually monitored by someone.



Figure 1. Left image shows a number of video surveillance cameras installed on a street light pole. Right image shows a surveillance control room with a wall of video surveillance monitors.

NOTE: The question is how often is someone actually watching from the other end of these cameras? In reality, a human operator can keep track of only a few video streams for some of the time.

The problems described previously can be alleviated to a great extent by incorporating intelligent, efficient video analysis and threat warning algorithms into the video surveillance system. In most cases, security personnel are interested in recognizing certain patterns of movement by certain types of targets or discovering certain unusual changes in the scene by viewing the video streams. The closer an algorithm can perform the intended surveillance functions automatically, accurately, and efficiently, the higher acceptance and usage it will enjoy in the video surveillance community. Obviously, many mundane movements and changes displayed on the monitors, such as wavering tree branches driven by strong wind and changing shadows due to sun movement, are to be ignored in general. Algorithms have been proposed to suppress such benign background distracters, including swaying trees in wooded scene (4) and roaring waves at beaches (5), with some success.

Human and vehicular movements are the primary foci of video surveillance personnel in general, even though most of these movements are usually benign and normal in the settings under surveillance. Instead of having to follow the tracks of all the movements, security personnel would be better served if they were only alerted to examine those tracks that were automatically flagged as suspicious movements by intelligent video surveillance algorithms. Based on the Dempster-Shafer theory of evidence framework, for instance, Snidaro et al (6) proposed a real-time trajectory clustering algorithm to learn common patterns of activity, and then detect any unusual trajectories that may need a closer scrutiny for behavior analysis and situation assessment. Sometimes, only human targets are monitored by the security personnel due to a

higher level of threat and suspicion associated with the presence of pedestrians in certain scenarios. A number of algorithms were proposed to detect human moving targets in video streams (7), including but not limited to those methods based on human gait motion (8), skin detection (9), and face recognition (10). On the other hand, some surveillance applications (e.g., monitoring highway traffic congestion using ground sensors [11]) or sensor limitations (e.g., too few pixels on human targets can be picked up by aerial sensors situated at high altitude [12]) may restrict the target set to vehicular objects only. Examples of existing vehicular tracking algorithms include those based on Projective particle filter (13) and rigid body target modeling (14) approaches.

In addition to human and vehicular movements, security personnel may also be interested in some abnormal changes in the scene, such as abandoned baggage, removed property, or oddly positioned objects. While these irregularities can be harmless at times, they may well be the indications of improvised explosive devices (IEDs), burglary, illegal parking, malicious intrusion, personnel injury, or structural damages that need immediate attention. For instance, Bhargava et al. (15) described a general framework to detecting an unattended baggage in forbidden areas, based on the temporal flow leading to the event. By backtracking and analyzing the recorded video stream, the true owner of the suspicious baggage may be identified and investigated. A number of approaches were proposed to perform and improve the detection of these movements and changes in video streams, some of which were summarized by Hu (16) and Morris (17) in their survey paper, respectively. Obviously, detecting the movements and changes alone may not be useful enough to understand what is happening and what the response should be. There is a need to recognize the underlining activity and behavior based on the movements and changes observed, as well as any prior or external information related to the given scenario. In recent years, semantic understanding and interpretation (18) of video events by machines—the automatic recognition of human behavior and activities (19, 20), in particular—has emerged as an important but challenging multi-disciplinary research area.

Due to the security requirements of military facilities and the growing threats of terrorists, military physical security personnel are increasingly worried about potential intrusions by unauthorized personnel and vehicles, as well as unconventional attacks by hostile elements. Human and vehicular movements at certain times, areas, speeds, and directions can be deemed as suspicious, while object-abandoning, box-carrying, picture-taking, and wall-scaling are abnormal activities that are worth a closer look. With the objective of enhancing the physical security of critical facilities, the Force Protection Surveillance System (FPSS) has been developed using an intelligent Moving Target Indication (MTI) algorithm.

FPSS processes input video streams, tracks the targets of interest in the scene, and reports critical information, such as potential intrusions and suspicious activities, to the security personnel immediately. Early discovery of these suspicious objects or events may enhance the safety and security of the protected community. For example, prompt detection of an abandoned package near a parking lot may avert a deadly IED attack by defusing the bomb in a timely manner.

Many suspicious objects or events may occur after night fall and will not be detected by human eyes or electro-optical cameras operating in the visible spectrum. Therefore, the FPSS was designed to process both color and forward-looking infrared (FLIR) imageries, so that it may perform well around the clock and under visually challenging conditions, including foggy or smoky scenes. On the other hand, FLIR imagery has its own share of phenomenological challenges, which include reversed polarity of the same targets in different seasons, varying target-to-background contrast as the ambient temperature changes over the course of a day, and often spotty heat emissivity from a given large target. The FPSS algorithm is able to achieve satisfactory tracking performances by overcoming some of these FLIR difficulties in particular.

The backbone of the FPSS algorithm is a set of disjoint intermediate background models that are intelligently structured to form accurate and dynamic representations of the most recent scenes. Using an adaptive background subtraction method, the signatures of moving targets are first captured and then enhanced through a set of image filters to reduce noise and ambiguity. The next movements of these targets are reasonably estimated using a set of kinematic predictors, so that a persistent track can be maintained. The details of these methods are provided and explained in the next section, while section 3 describes the experimental setup and results pertained to the FPSS algorithm. Some concluding thoughts are given in the final section of this report.

2. FPSS algorithm

After the FPSS algorithm is initiated, it accepts image frames from an input video stream for its preprocessing stage. In real life scenarios, sometimes the input video stream may not contain the anticipated video signals due to a foggy scene, a blocked field of view, or failures in camera, transmission line, or video capture card. Therefore, it is prudent to first examine whether or not the input image frames contain any discernable features by computing the mean and standard deviation of all pixel values of each input frame. If blank or near-blank image frames are detected, these frames will be discarded from further processing and a warning alert is sent to the human operator. Otherwise, the image frames will be uniformly down-sampled to speed up the tracking process, while maintaining an acceptable level of tracking performance. The amount of down-sampling is set by the human operator, which is often dependent on the original resolution of input images, the number of pixels on the smallest target of interest, and the reasonable trade-off between speed and accuracy for a given tracking task and condition.

The image frames acquired from different cameras, especially those with multiple-byte value per pixel, may vary significantly in terms of dynamic range and minimum value of the pixel values. Some cameras also automatically adjust their dynamic range and brightness periodically. To reduce this kind of variability and disturbance, FPSS normalizes the dynamic range of all input images by applying histogram equalization on them. Optionally, median filtering can be applied

to reduce salt-and-pepper noise that may exist in the original input images or is somehow induced by the previous preprocessing steps. Given the computationally expensive nature of median filtering, it is recommended for fairly noisy input image streams only.

At the end of the preprocessing stage, the resulting images are fed through an adaptive background modeling module, a target detection module, and a target tracking module sequentially. These three modules are described in the following subsections.

Background modeling

In order to detect moving targets in a video stream, typical approaches (21) may include background subtraction (22), optical flow analysis (23), moving energy analysis (24), and temporal differencing (25) methods. Among these approaches, background subtraction is the most popular method due to its simplicity in implementation and efficiency in computation, especially when the input video streams are captured by fixed cameras. Compared to other approaches, the background subtraction method provides the most complete feature data, but this method is rather sensitive to environmental changes in a dynamic scene and not suitable for the applications where the cameras are moving. On the other hand, optical flow method is capable of detecting independent moving targets in video streams captured by moving cameras, but its computational complexity and requirements may exclude this method from real-time, full-frame video surveillance applications. To the opposite of the background subtraction method, moving energy analysis is robust in detecting certain targets buried in a complex and dynamic scene, but this method is unable to extract exact feature data of the moving targets. Similarly, the temporal differencing method is also very adaptive to complex and dynamic scenes, but it is not capable of extracting all relevant feature data, in general. Because the surveillance cameras for force protection applications are usually installed at fixed locations, and the scenes around military installations are usually not very complex or dynamic, the background subtraction method was chosen and implemented for the FPSS algorithm.

The FPSS background modeling and subtraction process is depicted in figure 2. Each input image is first filtered by a stability mask and then channeled through four imager buffers of equal size and depth. The images in Buffers 2 and 4 are used to generate Background Models 1 and 2, respectively. Instead of creating it anew from Buffer 4, Background Model 2 can also be obtained from a buffer of models that is continuously replenished by the outgoing representations of Background Model 1. By subtracting the next input frame from these background models, we can obtain two separate difference images. A difference-product image (DPI) is then generated by multiplying these two difference images pixel by pixel. In addition to its function of affecting the properties of stability mask from time to time, the DPI is also essential to the subsequent detection and tracking processes of FPSS.

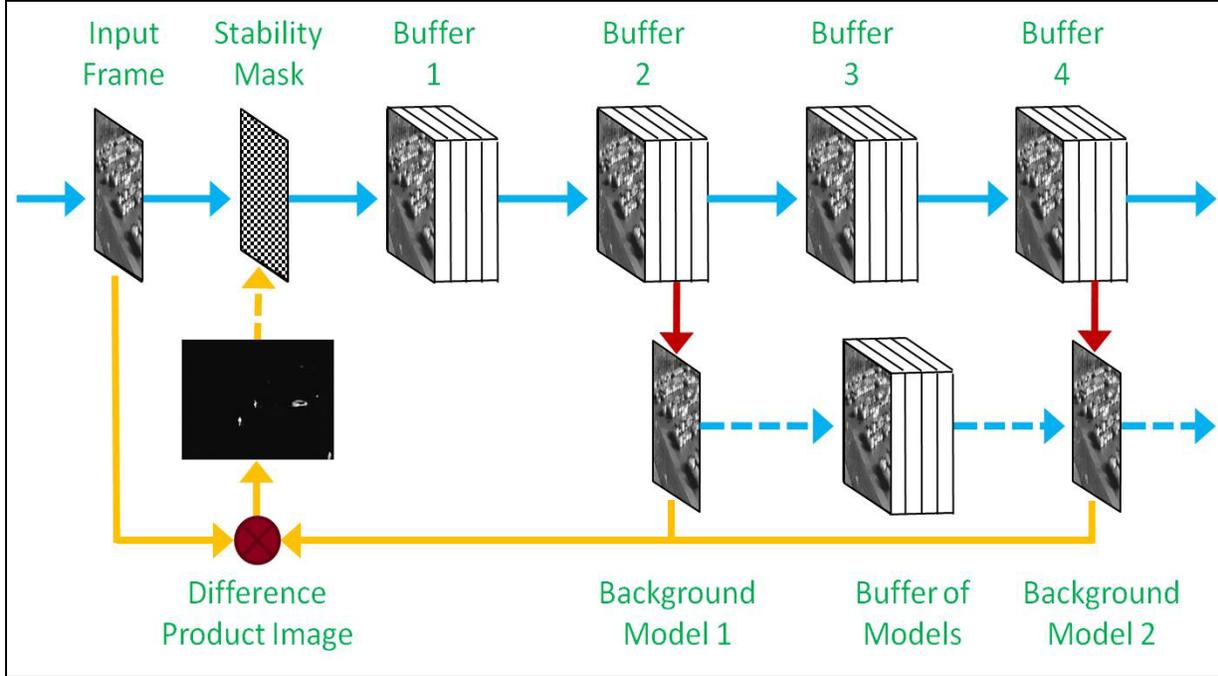


Figure 2. The schematic diagram of the background modeling process in FPSS.

To begin the background modeling process, the first successfully preprocessed input image frame is used to fill up all image buffers and to become the initial background models. For each of the subsequent input image frames, a simple frame registration procedure is used to reduce any potential jitter effects incurred by shaking cameras. A number of small and evenly distributed image patches are defined, each of which is allowed to move within a predefined 2-D neighborhood on the input image. By moving a patch within its neighborhood, a better match region between the input image and the current background model around that location may be found. The horizontal and vertical movements associated with the best match, (m^*, n^*) , for a given patch are computed as follows:

$$(m^*, n^*) \text{ for } \min \sum_{m=-s_x}^{s_x} \sum_{n=-s_y}^{s_y} \sum_{x=g}^{g+p_x} \sum_{y=h}^{h+p_y} |I_i(x+m, y+n) - I_b(x, y)| \quad (1)$$

where $I_i(x, y)$ and $I_b(x, y)$ is the pixel intensity value of the input image and background model at location (x, y) , s_x and s_y are half of the horizontal and vertical neighborhood size that each patch is allowed to roam, and p_x and p_y are the horizontal and vertical size of each patch, while g and h is the horizontal and vertical shift of a given patch from the upper-left corner of the input image, respectively. The values of g and h are dependent on the number, size, and distribution pattern of the patches, while the total size of all patches should not cover a significant portion of the input image.

When a noticeable and consistent movement is recorded by the majority of the patches, then a frame-wide jitter or movement may have occurred. Based on this consistent movement, the input image is shifted accordingly in order to achieve better registration with the current background model. This method works well with small jitters, such as those incurred by wind, but is not suitable for drastic scene changes caused by pan-tilt-zoom operations of the camera. These drastic changes are handled by the background updating procedure, which often requires a complete flush of the current background model and replacing it with the new scene.

Each of the jitter-free images is then subtracted from the two background models to produce a DPI. When the two background models are identical to the first preprocessed input image, the resulting DPI is merely a difference image with squared difference value for each pixel. The product term introduced in this step is useful for the subsequent target detection module, because bright blobs will be generated for all moving targets regardless of the polarity of their original brightness with respect to their immediate background. Although taking the absolute value of each pixel in the first difference image may achieve similar effects in the beginning, the DPI exhibits much better characteristics when the two background models later evolve into two background representations that are clearly disjointed in time. For instance, transient noise will be better suppressed in the DPI than in an absolute-difference image.

Typically, the preprocessed input image frame contains a mostly stable background with a number of small but volatile areas caused by moving objects and other transient events. In order to prevent these rapidly changing foreground pixels from ruining the background model, a stability mask is used to filter out all unstable pixels from the input image frame. Supported by the information provided by the DPIs, this stability mask looks for significant intensity changes based on a predefined threshold of variability and maintains a record of stability index at each pixel location. Only the stable pixels on a given input image frame are fed to Buffer 1, while the once-stable but now active pixels are blocked and substituted by the corresponding stable pixels available from Buffer 1. Figure 3 clearly shows the positive effects of the stability mask in generating stable background models. Without the stable background models, it will be much harder to detect and extract legitimate moving objects in the scene, while additional false alarms will likely be generated.



Figure 3. The effects of stability mask on background models. Left image shows that active foreground pixels (identified with red ovals) could seep in and ruin the background models when the stability mask is deactivated. Right image shows a much cleaner background model when the stability mask is activated.

Each newly arrived set of pixel values replaces the oldest frame in Buffer 1, while the oldest frame of Buffer 1 becomes the newest frame in Buffer 2. The same mechanism of first-in first-out (FIFO) frame-shift and update is applied to all image buffers continuously. The number of image frames stored in each buffer is defined by the FPSS user, with 3–10 frames as a reasonable number under most circumstances. The role of Buffer 1 is merely a time-delay buffer, so that there is a noticeable gap in time—and potentially in content—between the current input image and the image frames in Buffer 2. Background Model 1 is derived from the images in Buffer 2, which can be as simple as taking the average of all images in Buffer 2. Similar to Buffer 1, Buffer 3 is just another buffer to separate Buffer 2 and Buffer 4 in time. Background Model 2 can be obtained by either processing (e.g., averaging) the images in Buffer 4 or drawing from the Buffer of Models supplied by Background Model 1. The same background modeling structure depicted in figure 2 can be extended to include four or a larger even number of background models. The extended structure is able to achieve even more stable background representations and higher target enhancement capabilities at the expense of additional computational resources.

One of the advantages of using multiple disjoint background models to generate a DPI is that the problematic “trailing effect”, which is often associated with background subtraction method, can be suppressed effectively; because those gradually fading trails carved out by the moving objects are now showing up in different parts of the two difference images, they are likely to diminish or disappear when the DPI is formed, as demonstrated in figure 4. Another advantage of this method is that the trails are now clearly detached from the moving objects, which allows the subsequent target detection module to estimate the size and location of those movers more accurately. With improved estimation in target size and location, the target tracking module may then perform better in motion estimation and track maintenance.



Figure 4. Enhancement of target signatures and suppression of trailing effects and noises are achieved simultaneously in a DPI. Two difference images on the left show the trailing effects and random noises in different places, which are suppressed in the resulting DPI. Signatures of legitimate moving targets appear in the same locations on both difference images, thus, they are enhanced in the DPI through the multiplicative process.

An even number of background models are needed in the formation of DPI to address the problem of target polarity, which is a common target detection problem, in general, and a FLIR target detection problem, in particular. As the seasons and ambient temperature change, the same type of moving targets (e.g., walking humans) observed in FLIR imagery may assume different polarity of pixel intensity with respect to their immediate background due to automatic gain control (AGC) of the camera. Figure 5 shows a pair of FLIR images that exhibit polarity change in human signatures during different seasons of the year. A similar problem occurs in the visible imagery when the illuminance of moving targets is flipping between or occupying in both the upper and lower sides of the illuminance of their immediate background. Using a single difference image or a DPI computed with any odd number of difference images to detect the moving targets will have to pick the locations with both positive and negative values simultaneously and appropriately, which is not always easy or straightforward. This problem is alleviated, however, simply as a by-product of forming the DPI using an even number of difference images.



Figure 5. Polarity changes are common in FLIR cameras equipped with automatic gain control. Left image shows that human signatures are clearly brighter than their background in a typical winter afternoon. Right image shows that human signatures are much darker than their background in the same parking lot during summer time.

Target detection

The function of the target detection module is to estimate the size and location of all moving targets in the current input image frame based on the information received from the background modeling module. As described in previous subsection, a DPI is generated by multiplying two or any even number of difference images pixel to pixel. The surface of the resulting DPI is often rough and plagued with gaps. Therefore, a morphological operation is used to remove small spikes and to fill up small gaps on the DPI. Usually, this smoothing process is achieved by a sequence of opening and closing operations, with a window size of 3×3 or 5×5 pixels in FPSS, based on the typical target sizes observed in the FPSS dataset.

Although the morphological operation does make the surface of DPI smoother and more connected, it does not necessary enhance the centroid and overall silhouette of the moving targets. A pyramid-means method is used in FPSS to increase the “blobiness” of moving targets by replacing each pixel value of the morphologically filtered DPI with the average of three mean pixel values that are computed based on the surrounding three different-sized rectangular areas. Figure 6 shows the DPI (upper-right) of a given input image frame (upper-left), as well as the post-processed DPI after the morphological (lower-left) and pyramid-means (lower-right) operation, respectively. It is obvious that the morphological filter has removed all noticeable speckle noises caused by spurious movements, while improving the structure of the target signatures. The pyramid-means operation has significantly increased the “blobiness” of the target signatures, especially those pertinent to human targets in this example.



Figure 6. The upper-left image is an image frame in a FLIR video sequence. The upper-right image is the corresponding DPI with some small but clearly visible noises. These noises are removed after the morphological operation (lower-left). The “blobiness” of moving targets is increased in the lower-right image through a pyramid-means method.

The actual moving target detection process begins with finding the brightest pixel on the post-processed DPI, which is usually associated with the most probable moving target in the given input frame. The size of this target is estimated by finding all the surrounding pixels that are deemed as connected to the brightest pixel. This connectivity is determined by the existence of an unbroken path from a given pixel to the brightest pixel, where each pixel in this path must satisfy a predefined or adaptive threshold of brightness. In FPSS, the connectedness is explored by performing multi-directional recursive searches emanated from the brightest pixel. The farthest extents of those searches are used to establish the horizontal and vertical boundaries of the current target. For the sake of simplicity, a rectangular area defined by these boundaries is considered as the shape and size of the given moving target. In figure 7, for example, the first moving target detected in this input frame is a fast-moving car that is tagged as Target 1 and delimited by a blue bounding box.

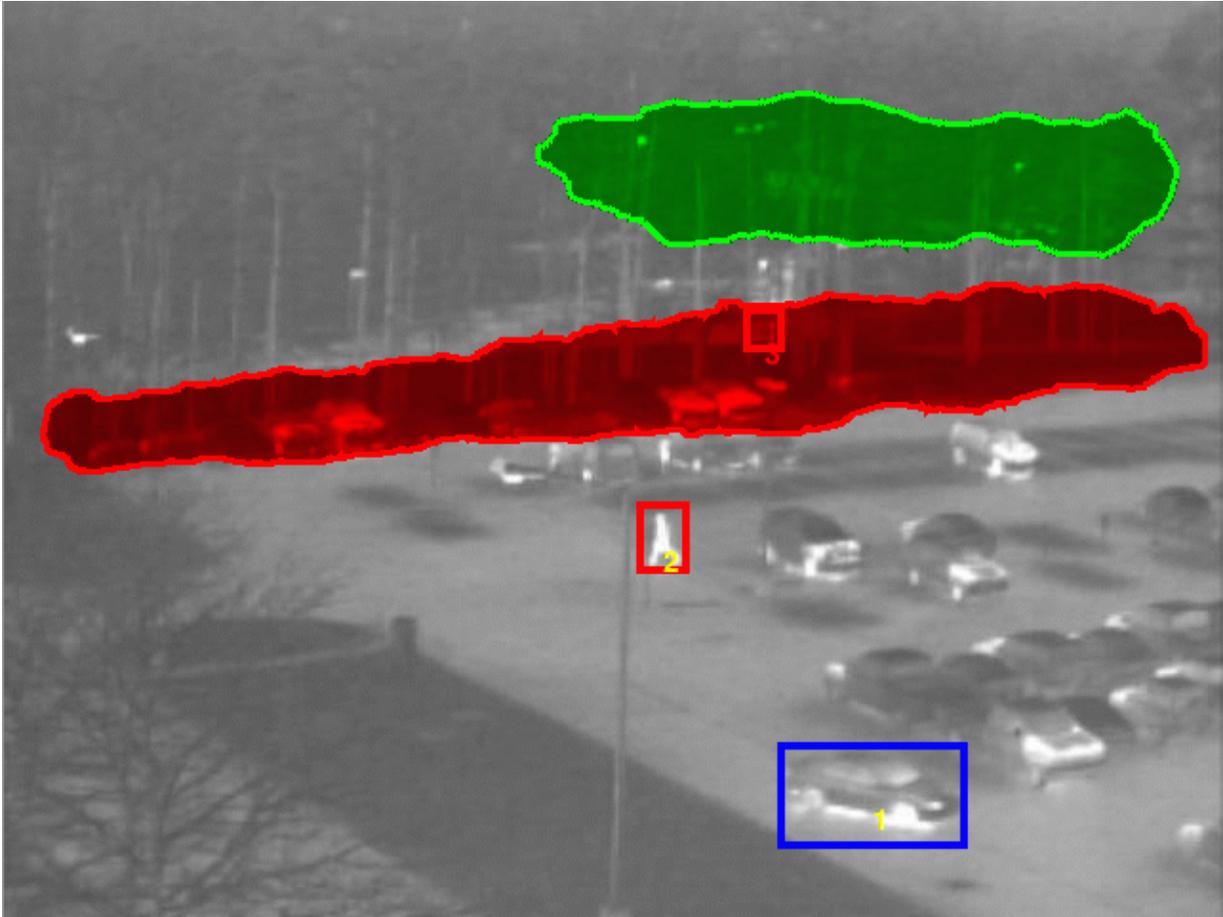


Figure 7. Three moving targets are detected on this input frame, which are tagged as Targets 1, 2, and 3, respectively. The size of each target is delimited by a rectangular bounding box and the potential type of target is represented by the color of the box. The green area is a “don’t care” zone, while the red area is a “critical” zone.

After the first moving target is detected, all the pixels within that rectangular target-sized area are flattened to the minimum value of the post-processed DPI, so that none of these pixels will be considered again as another potential moving target. The detection process continues by finding the next brightest one among the remaining pixels and repeating the process until all the pixels are flattened, a predefined number of detections are obtained, or other user-defined parameters have prevented further searches. These parameters may include the minimum and maximum size of potential targets, the proportion of overlapping area allowable between adjacent targets, and the “don’t care” area, in which all detections should be ignored. In figure 7, the green area is a “don’t care” zone that covers a public road outside of the main gate; the traffic activities in that area are ignored by FPSS. On the other hand, the red patch is a wooded area just inside of the main gate, which is a “critical” zone, where movements are subjected to a higher level of scrutiny. A total of three moving targets are detected on this input frame, one of which is found in the critical zone.

As shown in figure 8, the FPSS can also perform change detection by focusing on stationary changes between the current input image and a previously saved reference background. The reference background can be refreshed at any time, or a number of reference backgrounds can be created and saved for different times and scenes. In addition to detecting abandoned objects, this function is also very useful in detecting purposely move-and-stop objects, such as snipers in a wooden area. The change detection and moving target detection can be operated concurrently or separately in FPSS, and they are controlled by two independent sets of parameters.



Figure 8. FPSS change detection capability. A man left a suitcase on the traffic island adjacent to parking lot (lower-right and middle images). FPSS detected the scene change and highlighted the abandoned object within a few seconds (upper-left image).

The results of change detection and/or moving target detection are reported to the user via a graphical user interface (GUI). As shown in figure 9, the GUI of FPSS allows a user to enter or modify a number of parameters related to the file directories, input images, potential targets, tracking characteristics, background modeling, and jittery control. Furthermore, the user may define, activate, deactivate, and remove any “don’t care” zone, “critical” zone, and trip wire by using this GUI, as well. The detected moving or changed targets are annotated or highlighted over the input image frame for easy understanding. The tag number, location on the image, size in pixels, and activation strength of all detected targets on each input frame are displayed at the bottom-right corner of this GUI. This GUI can also be used to extract image chips based on the location and size of all detected targets for other security surveillance applications, including a profiling sensor for human detection.

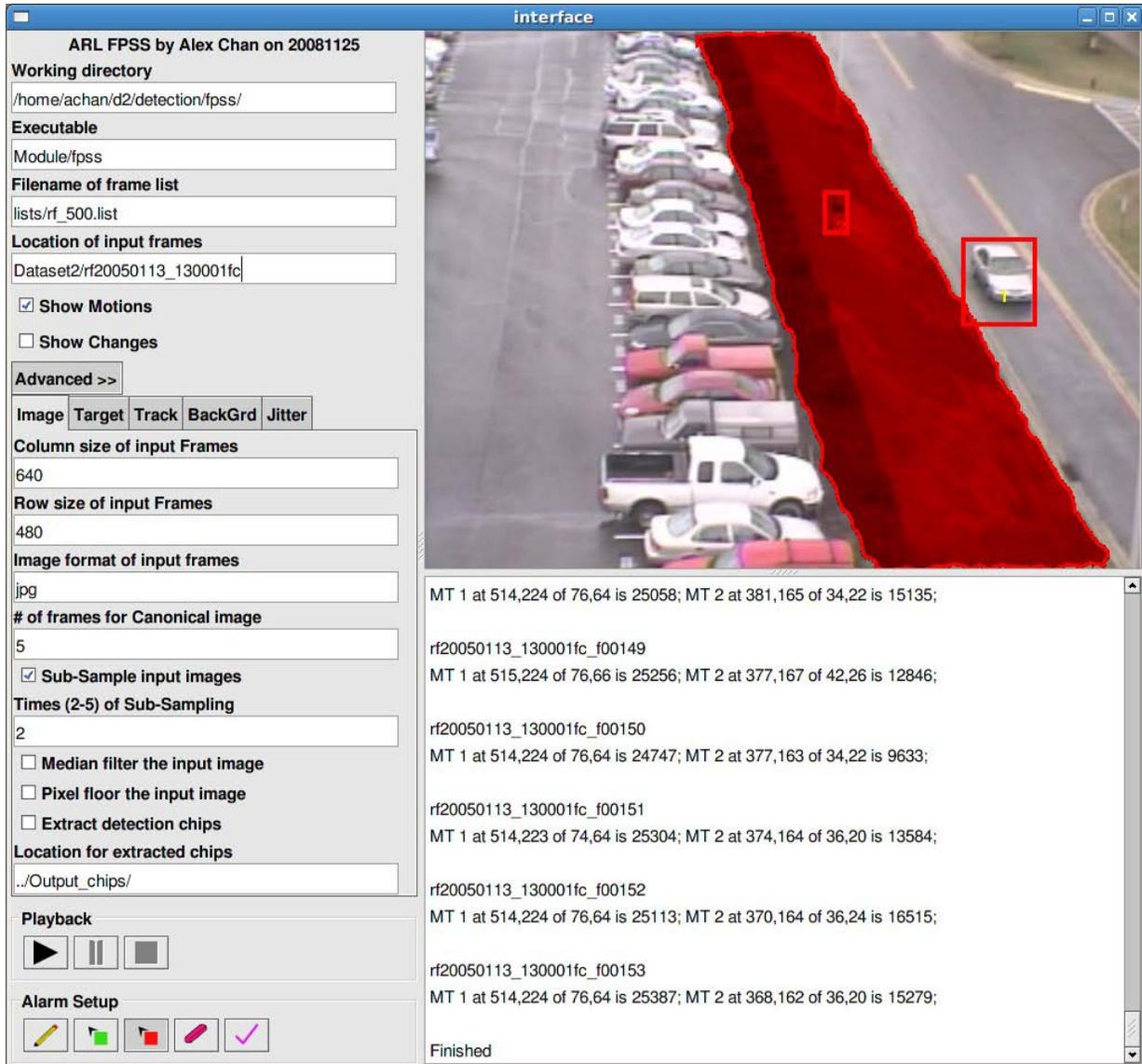


Figure 9. The FPSS graphical user interface.

Target tracking

The target tracking module uses the moving target information, extracted by the target detection module over a period of time, to build and maintain the track of each moving object. In order to build a meaningful track, it is required that a noticeable moving target must appear on multiple contiguous frames in a video sequence. This requirement may not be met when the target is moving across the field of view of the camera at a very short range and/or a very high speed, or when the camera is operated at a very low frame rate, the target is occluded for an extended period of time and/or behind a very large obstacle, or a combination of these and other detrimental factors.

Assuming the aforementioned requirement is met, the target tracking module in FPSS uses previous locations, velocity, and target size of a moving target to predict the destination of its next movement. Interestingly, as depicted in figure 10, most of the moving targets follow a rather straight traveling path (presumably the shortest and/or most feasible path) and a rather constant velocity most of the time. The perceived speed within the field of view also tends to positively relate to the observed size of the moving targets, either because the movers are traversing nearer to the camera or the bigger vehicles are moving faster than the smaller people. Based on these observations, the next destination of a moving target can be predicted fairly accurately. This is achieved by generating and updating a destination probability density map, in which the highest probability is assigned to the anticipated destination under an unchanged velocity. The spread of this destination map is proportional to the size of the moving target, because larger moving targets are more likely to end up in a broader span and range of destination. The shape and size of this destination map is also influenced by the velocity history of the moving target; for instance, a small elliptical destination map is reasonable for a slow and straight movement.

Charting the track of a single and open moving target is generally less of a challenge. Quite often, there are several moving targets of different sizes and velocity in a given proximity and their paths may cross or mix with each other. Under this circumstance, a moving target detected at a given location needs to be examined and compared with other targets in the area to avoid inadvertently mixing the tracks and wrongly tagging the targets. In addition to the target size and destination map information, a template match method is also used to further differentiate the moving targets from each other. Unfortunately, this approach does not always work; for instance, some people may look identical from a FLIR camera on the rooftop. Furthermore, when multiple groups of people merge, regroup, and then split again, tagging error is very likely to occur unless there are very distinctive features available to tell them apart from distance.

Another problem of track maintenance occurs when the moving targets become stationary for an unspecified amount of time. Sometimes a walking target may stop for a minute and then start moving again. In this case, the tracking algorithm should wait for a while and not create a new track for the same target when it starts moving again. On the other hand, if a vehicle comes into a parking lot and the driver leaves the vehicle soon after that, then the track for the vehicle may need to be terminated, and a new track created for the driver who is walking away. FPSS has different parameters to define temporary stopped and permanently gone instances. Targets that are deemed to have moved out of the scene for a period time are labeled gone and their tracks terminated.

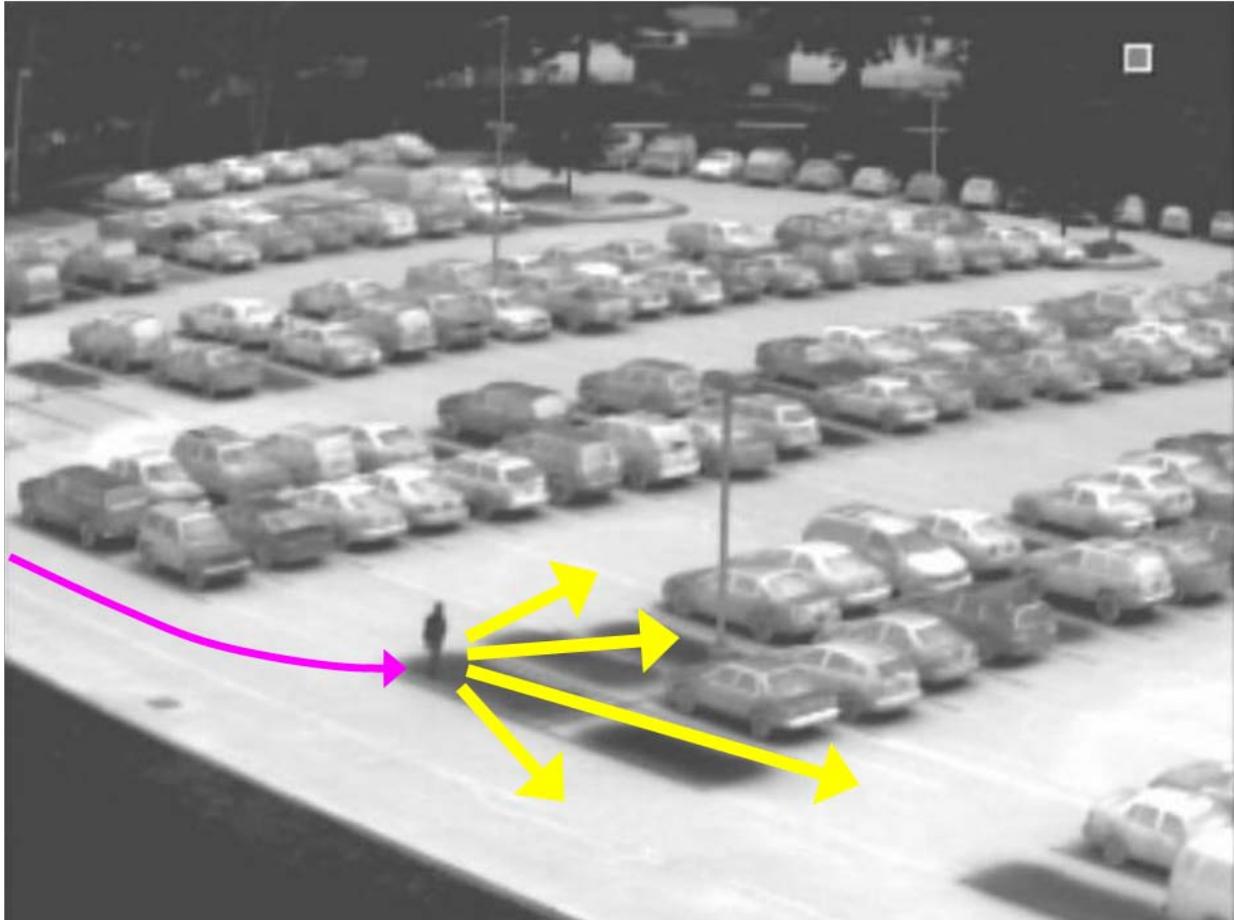


Figure 10. Using the track history (purple arrow) of a moving target, the FPSS predicts the likely directions and locations (yellow arrows) of the target in the near future.

Efforts are being made to improve the FPSS so that it may detect suspicious behavior and activities based on tracks analysis. For instance, if a vehicle comes and parks for a long time and no driver is seen leaving the vehicle, that vehicle may be considered suspicious and some additional actions may be necessary. Similarly, when one or more humans are detected as moving aimlessly in a large parking lot for a period of time, it may be worth a closer look to find out whether or not they are loitering, disoriented, or doing some other suspicious activities.

Obviously, there is a need to develop robust algorithms to detect and categorize the observed movements into more user-friendly semantic descriptions, such as “loitering,” “loss of direction,” “hijacking,” and “illegal parking.” These algorithms may rely on a sensible and effective set of rules and efficient data mining procedures to augment current tracking results with prior or external information, including functional scene descriptions, expected movements in a given area, rules and prohibitions in parking, speed limit, time of the day, weather condition, threat level of the installation, special occasion, and other related factors.

Extended time coverage over an area of interest is crucial in understanding the normal pattern of activity in that area. Figure 11 shows the track history for a given area over the course of a few hours. As expected, most of the traffic occurred along the road and sidewalk. However, there are a number of suspicious tracks that crisscross the road to a steep slope, as well as tracks entering and exiting a small tent on the grassy patch between the road and sidewalk. These suspicious movements are strangely deviated from the established norm for this area. Therefore, an investigation is aptly warranted.



Figure 11. Analyzing the track history (red arrows) is useful in establishing a normal pattern of movements and detecting suspicious movements in a scene.

3. Experimental Results

In order to have a quantitative assessment on the FPSS algorithm, we examined its performance using a large and realistic set of parking lot video events collected from a fixed rooftop camera. The resulting FPSS detections were compared to the corresponding moving target ground-truth information that was manually generated by human experts. The dataset and the corresponding FPSS results are further described in the following subsections.

Dataset

The dataset used in this performance assessment is referred to as the Second FPSS Dataset in this report, which was collected using the Thermal Vision Sentry Personnel Observation Device (POD) manufactured by the FLIR Systems. Shown as the left image in figure 12, the Sentry POD consists of a long-wave infrared (LWIR) uncooled microbolometer and a color visual camera, which are bore-sighted and integrated in a sealed enclosure. The microbolometer has a focal plane array (FPA) resolution of 320×240 pixels and a field-of-view (FOV) of $24^\circ \times 18^\circ$, while the color visual camera is capable of producing 460 NTSC TV lines and its FOV was set to about 24° horizontally, as well. Instead of producing calibrated pixel intensities that are consistently tied to the actual surface temperature of the scene observed, the Sentry POD constantly stretches the observed thermal profile across its intensity spectrum to maximize the contrast of the imagery outputs. Due to this AGC function of the Sentry POD, the resulting LWIR signatures for a constant-temperature object, such as a human, may change based on the background temperature in different scenes. The Sentry POD was installed on a rooftop and surveyed a parking lot nearby. The right image of figure 12 shows a typical view of the parking lot on a sunny day from the rooftop.

In the Second FPSS dataset, there are 53 concurrent pairs of color-FLIR image sequences, recorded at 10 Hz and stored in JPEG image format, totaling 71,130 image frames. A global linear transformation method was used to co-register each color-FLIR image pair to an acceptable level, while scaling the registered images to a common size of 640×480 pixels. Most of the scenarios in this dataset were actively staged by collaborators to represent a variety of suspicious activities and behaviors, such as a man abandoning a suitcase beneath a car, several people loitering around the parking lot, and passengers running away from a van. Figure 13 shows the color and FLIR images of a man soon before he abandoned his suitcase beside the trashcan next to the stairway. By comparing the lamp post at the upper-left corner of these two images, the color image on the left appears to exhibit a higher degree of warping at its corner area than that of the FLIR image.



Figure 12. The Sentry POD used in the FPSS data collection effort (left) and a typical view of a parking lot (right).



Figure 13. A pair of concurrent Color-FLIR images in the Second FPSS Dataset that was coarsely co-registered.

This dataset was collected in many separate occasions and under different weather conditions over a period of several months, hence it exhibits a wide range of variability and challenges. For the color sequences, there are common problems of shadows under the bright sun, headlight glare and windshield reflections at night hours, and the inability to detect and track the targets occluded by darkness, fog, and rain. For the FLIR sequences, some moving targets assume a temperature profile that is very similar to that of their immediate backgrounds, hence the target signatures are often faded into the background and become very difficult to detect consistently. Furthermore, there are many complex trajectories of moving targets in these scenarios, where the paths of people and vehicle are often obscured, crossed, split, grouped, and regrouped from time to time.

Accurately extracted ground-truth information is very important in providing a trustworthy basis of reference for any algorithm performance assessment. Using a GUI, the locations and target types of all moving targets in each frame in the FPSS dataset are manually but efficiently determined and recorded. As shown in figure 14, this GUI allows users to detect, track, classify, and record the targets of interest in a given image sequence with ease. There are five categories of target type defined in this GUI, namely, “person,” “vehicle,” “animal,” “unknown,” and “other,” where the “other” category can be specified differently on each frame as necessary, such as lawn-mover, motorcycle, or ignored-car. The information provided by the user for each sequence is saved to an ASCII ground-truth file, which contains the number of moving targets in each category and the corresponding X-Y coordinates for the center of each moving target in each frame. In hindsight, we should have also generated a bounding box to represent the size of each moving target and assigned unique name tags to differentiate all moving targets in a given sequence.

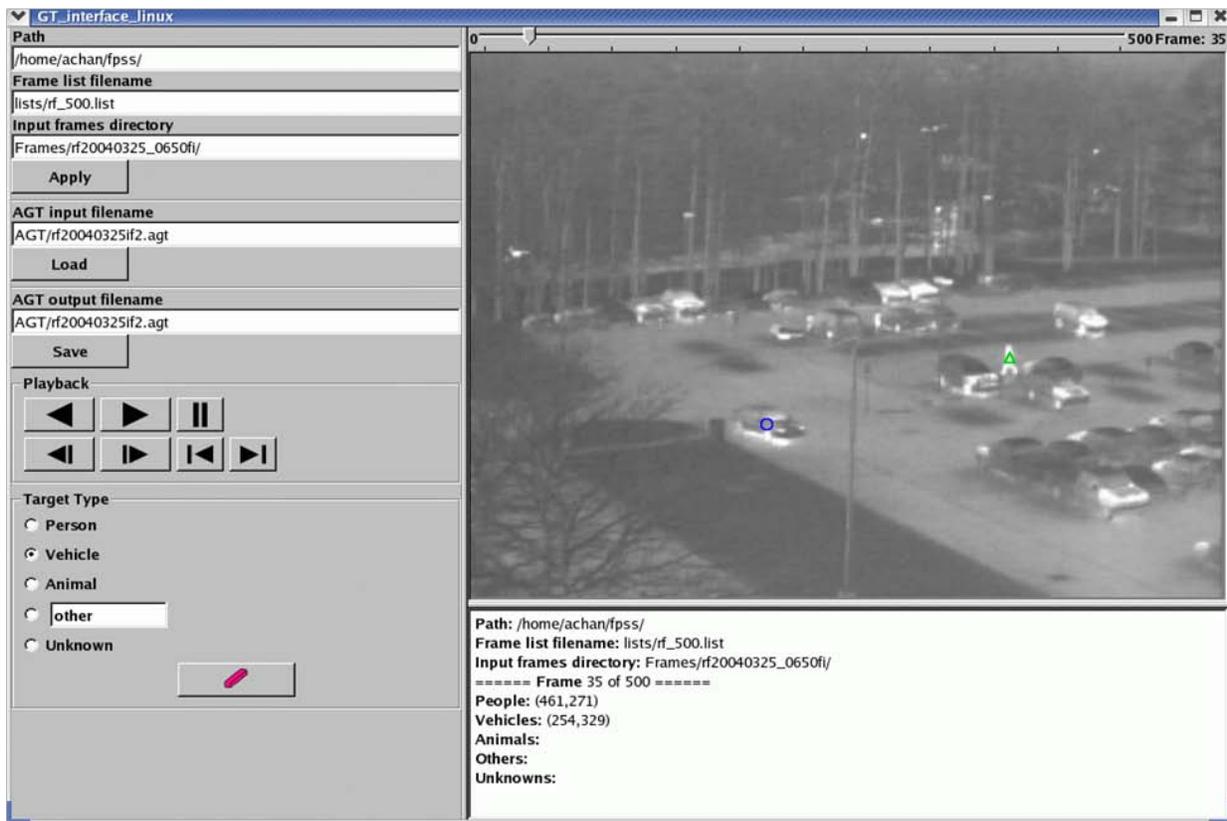


Figure 14. The graphical user interface used for the FPSS ground-truthing effort.

Performance and analyses

For a given video input sequence, the FPSS algorithm produces the location and estimated size of each potential moving target on each frame. An acceptance window is defined as a rectangle centered at a given FPSS detection with a dimension equal to the target size estimated by FPSS for that detection. This definition of an acceptance window is used here because the ground-truth size of moving targets is not available for the FPSS dataset, and the target size estimated by FPSS is deemed to be quite comparable to actual target size based on visual comparisons. If a ground-truth target falls within the acceptance window of any FPSS detection, it is declared as a hit. Multiple detections on the same target are counted as only one hit, but multiple ground-truth targets within the acceptance window of an FPSS detection are considered as a hit for each of those targets. Any ground-truth target that is not linked to any FPSS detection is considered as a miss, while a false alarm (FA) occurs when an FPSS detection cannot be associated with any ground-truth target for that frame.

For the initial detection runs in this experiment, the minimum acceptance threshold of FPSS was set to 500. In other words, only the pixels with a value of 500 or higher in the post-processed DPI were considered as potential target locations. At this setting, 63,017 of the 70,540 ground-truth targets in the 35,565 color images were associated with 47,059 of the 62,747 detections generated by the FPSS. This detection performance is translated into a hit rate of 89.34% or a miss rate of 10.66%. Given the 15,688 FA generated in this run, the FA rate can be expressed as $15,688/35,565 = 0.441$ per frame. When the minimum acceptance threshold of FPSS is raised, the number of detections is decreased, and so are the numbers of hits and FA. The resulting receiver operating characteristic (ROC) curve for the detection performance on the color sequences is shown as the top graph in figure 15, where a low hit rate of 14.63% is obtained for a small FA rate of 0.066 per frame. Depending on the perceived or actual penalty of missing an important target and the cost of processing an additional FA, human operators may choose different zones on this ROC curve to operate the detection system.

For the FLIR sequences, 66,951 of the 73,187 targets in the 35,565 FLIR images were correctly detected by FPSS when its minimum acceptance threshold was set to 500. In other words, the hit (miss) rate of 91.48% (8.52%) was achieved in this case, while the accompanying 9,284 FA resulted in a FA rate of 0.261 per frame. Because a number of detections were linked to more than one ground-truth target, there were only 62,619 detections generated by the FPSS in this case. By changing the minimum acceptance threshold, an ROC curve was also generated for the detection performance on FLIR sequences. The resulting FLIR ROC curve, which is shown as the bottom graph in figure 15, shows a hit rate of 29.01% at a FA rate of 0.032 per frame at its conservative end.

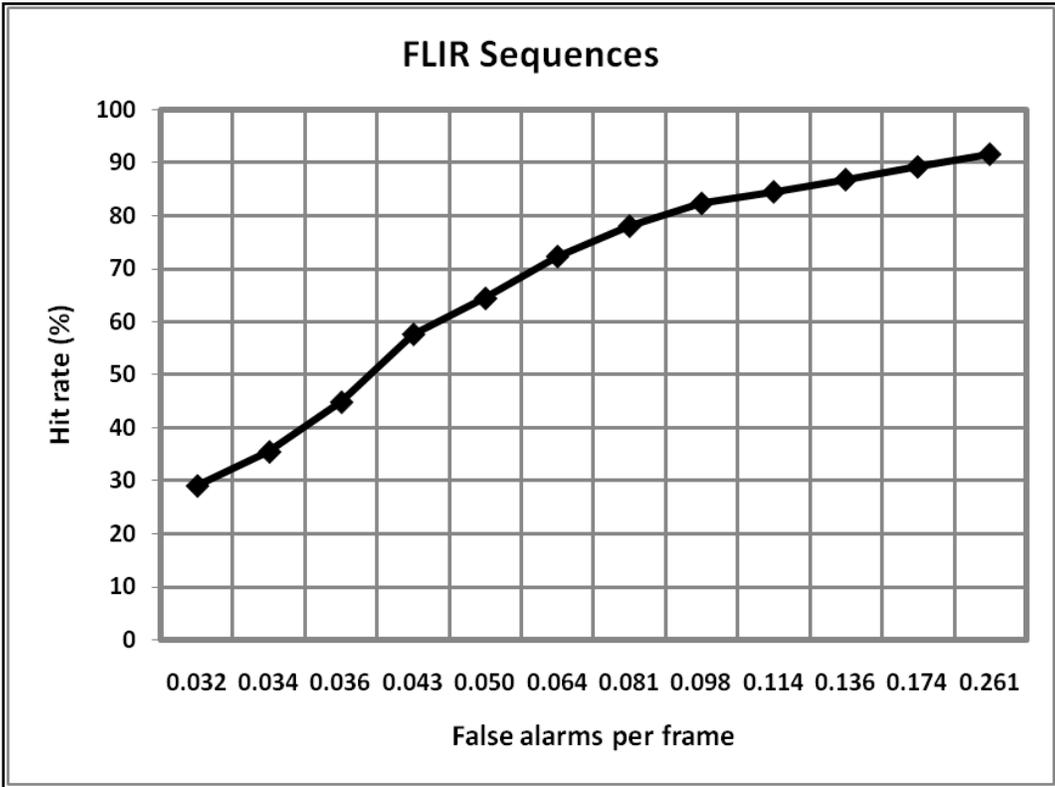
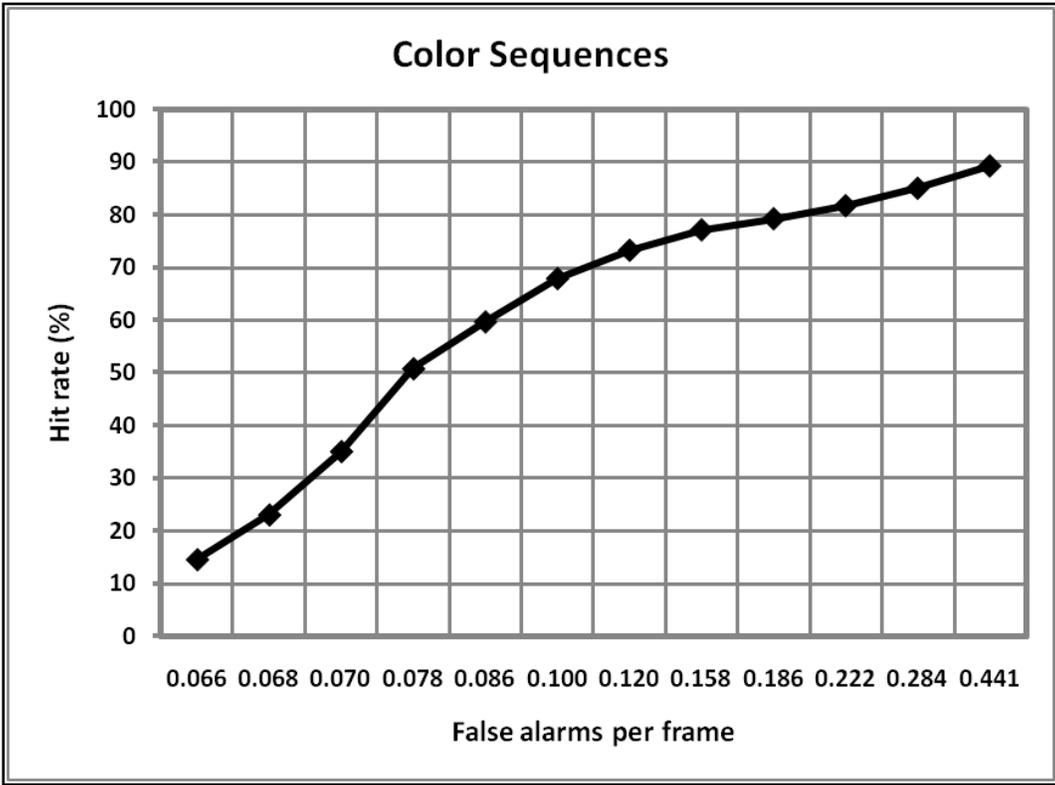


Figure 15. The ROC curve for the color (top) and FLIR (bottom) sequences, respectively.

The two ROC curves in figure 15 may look alike at the first glance, but the FLIR ROC curve indicates that the number of FA was reduced by more than 40% by performing detection on the FLIR sequences, while achieving slightly higher hit rates along the way. This significant reduction of FA in FLIR imagery can be attributed to the absence of nuisances associated with shadows, glares, and reflections, while these distractions often generate many FA in the color sequences. On the same number of image frames, the FLIR sequences also have 2,647 more ground-truth targets than the color sequences have, mainly due to the consistent presence of FLIR target signatures in night scenarios, whereas the color targets may appear in lighted areas but disappear into darkness periodically. Clearly, the FLIR sensor has certain advantages in security surveillance applications.

Because the color targets are more readily blended into their background due to darkness, fog, and clothing, they are more likely to be omitted during the ground-truthing process, as well. Although they were omitted from the ground-truth files, these evasive color targets may still be correctly detected by FPSS but wrongly labeled as FA due to the absence of ground-truth information. This is another potential reason for the higher miss and FA rates for the FPSS performance on the color sequences.

In order to avoid spurious tracks, FPSS ignores all transient moving objects that appear on just a few frames, even though some of these short-lived transient moving objects may, indeed, be legitimate targets registered in the ground-truth files. FPSS also starts to track a moving object only after its consistent appearance is firmly established. During this wait-and-see period, no detection is produced and all the ground-truth targets in these “tentative” frames were declared as misses by default. Although this “built-in” loss of detections negatively affects both the color and FLIR sequences, its effect is somewhat limited if the typical tracks of interest are sufficiently long as compared to the number of tentative frames.

Unlike some tracking algorithms, FPSS was designed for real-time performance and, hence, does not perform backtracking to recover potential misses and update the track based the detection results right before and after those misses. If a delay of several seconds in producing the tracking output is tolerable, then the performance of FPSS can be significantly improved by recovering many of the missed detections, including those omitted during the wait-and-see period in the beginning of each track, using newer information obtained after those misses. With the advantage of looking ahead and the resulting higher track consistency, many FA can be eliminated as spurious noises and inconsistent movements. As a result, the overall tracking performance is improved by achieving a higher detection rate and a lower FA rate.

4. Conclusions

Although human experts may be able to monitor one or two video surveillance cameras effectively by extracting meaningful information on moving and changed targets, recognizing the behavior and activities present in the scenes, and predicting the next movement or outcome with high confidence level, it is impossible to have enough human eyes and minds to process millions of video surveillance data continuously, efficiently, and economically.

The FPSS algorithm provides a practical solution that reduces the workload of security surveillance personnel, and increases the physical security of the monitored facilities by alerting the security personnel on moving and changed targets that satisfy some predefined characteristics and conditions. Based on these valuable and highly compressed alerts, the security personnel may respond to the potential threats in an informed and timely manner.

This report provides a rather detailed description and explanation of the FPSS algorithm in section 2, which is supplemented with the FPSS algorithm flowcharts in appendix A. The FPSS algorithm, as described in this report, was submitted to the U.S. Patent and Trademark Office in 2005 in order to be recognized as a valuable intellectual property. Eventually, United States Patent Number US 7,460,689 B1 titled “System and Method of Detecting, Recognizing, and Tracking Moving Targets” was granted in 2008, with the author of this report as the sole inventor of this technology.

The FPSS detection results provided in this report were the first set of quantitative measurements on the performance of FPSS. Using a large, ground-truthed, and security-oriented video dataset, the detection results of FPSS were found to be comfortably high and reliable, detecting about 90% of all known moving targets at a rather low false rate for both color and FLIR sequences. With additional research and improvements, such as short-term backtracking, long-term track analyses, and multi-sensor fusion, the FPSS may be further enhanced in terms of tracking accuracy and robustness in event characterization.

5. References

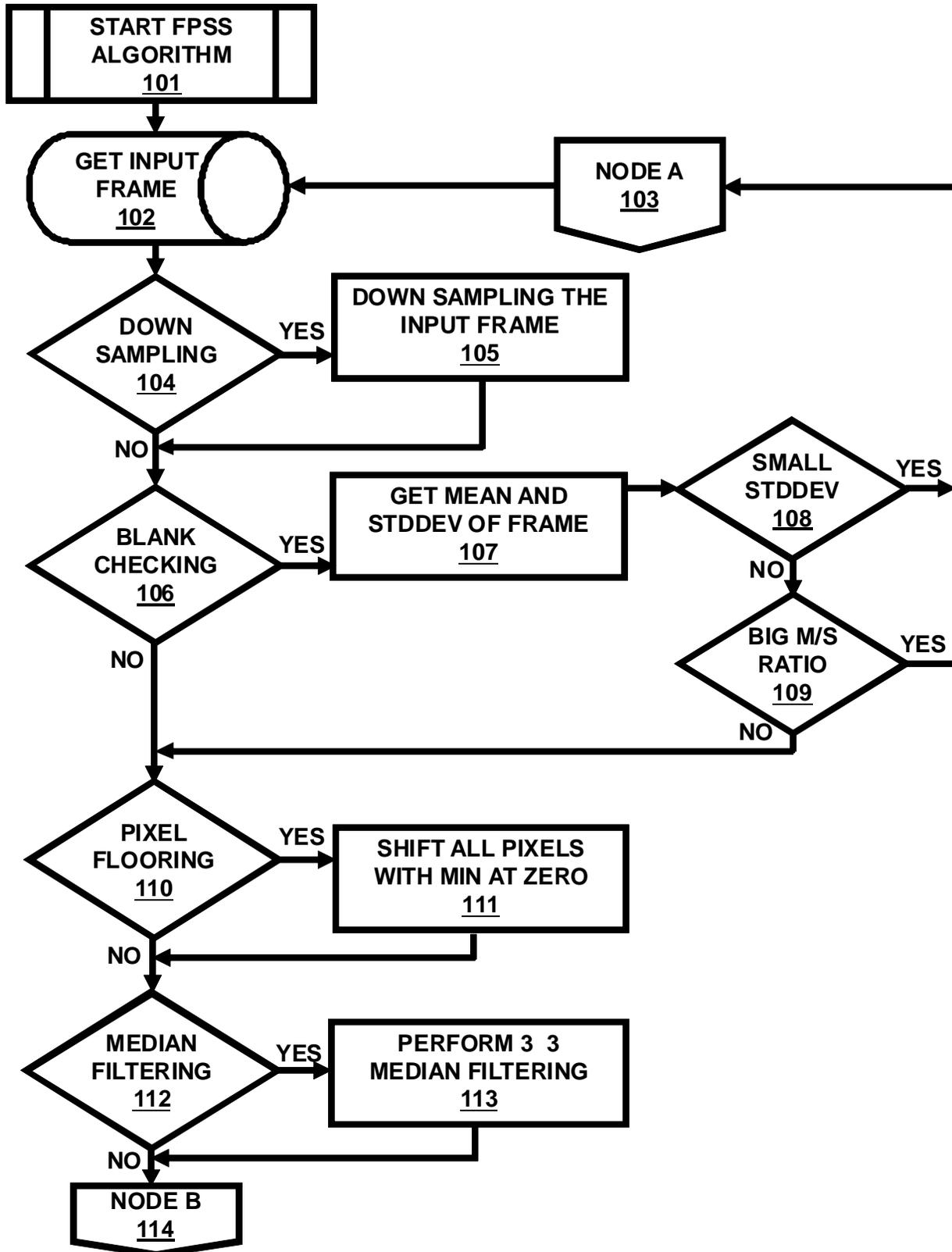
1. Ball, K.; Lyon, D.; Wood, D. M.; Norris, C.; Raab, C. A Report on the Surveillance Society. *Proc. 28th Int. Data Protection and Privacy Commissioners' Conference*, London, Sep. 2006.
2. Sears, C.; Pylyshyn, Z. Multiple Object Tracking and Attentional Processing. *Canadian Journal of Experimental Psychology* **2000**, *54*, 1–14.
3. Cavanaugh, P.; Alvarez, G. Tracking Multiple Targets with Multifocal Attention. *Trends in Cognitive Sciences* **2005**, *9* (7), 349–354.
4. Li, C.; Jiang, N.; Si, J.; Abousleman, G. P. Robust Target Detection and Tracking in Outdoor Infrared Video. *Proc. Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 1489–1492, Las Vegas, April 2008.
5. Yu, Z.; Chen, Y. A Robust Motion Detection Algorithm for Complex Background Using Statistical Models. *Proc. IEEE Conf. Robotics, Automation and Mechatronics*, pp. 362–367, Chengdu, China, Sep. 2008.
6. Snidaro, L.; Piciarelli, C.; Foresti, G. L. Activity Analysis for Video Security Systems. *Proc Int. Conf. Image Processing*, pp. 1753–1756, Atlanta, Oct. 2006.
7. Chellappa, R.; Aggarwal G. Video Biometrics. *Proc. Int. Conf. Image Analysis and Processing*, pp. 363–370, Modena, Italy, Sep. 2007.
8. Kusakunniran, W.; Wu, Q.; Li, H.; Zhang, J. Automatic Gait Recognition Using Weighted Binary Pattern on Video. *Proc. Int. Conf. Advanced Video and Signal Based Surveillance*, pp. 49–54, Genoa, Italy, Sep. 2009.
9. Zhou, B.; Gu, Y.; Li, B.; Zhang, G.; Tian, T. A Practical Algorithm for Exception Event Detection for the Home Video Security Surveillance. *Proc. Int. Conf. on Info-tech and Info-net*, pp. 202–208, Beijing, China, Oct. 2001.
10. Kage, H.; Seki, M.; Sumi, K.; Tanaka, K.; Kyuma, K. Pattern Recognition for Video Surveillance and Physical Security. *Proc. Int. Conf. on Instrumentation, Control and Information Technology (SICE 2007)*, pp. 1823–1828, Sep. 2007.
11. Kanhere, N. K.; Birchfield, S. T. Real-Time Incremental Segmentation and Tracking of Vehicles at Low Camera Angles Using Stable Features. *IEEE Trans. Intelligent Transportation Systems* **Mar. 2008**, *9* (1), 148–160.

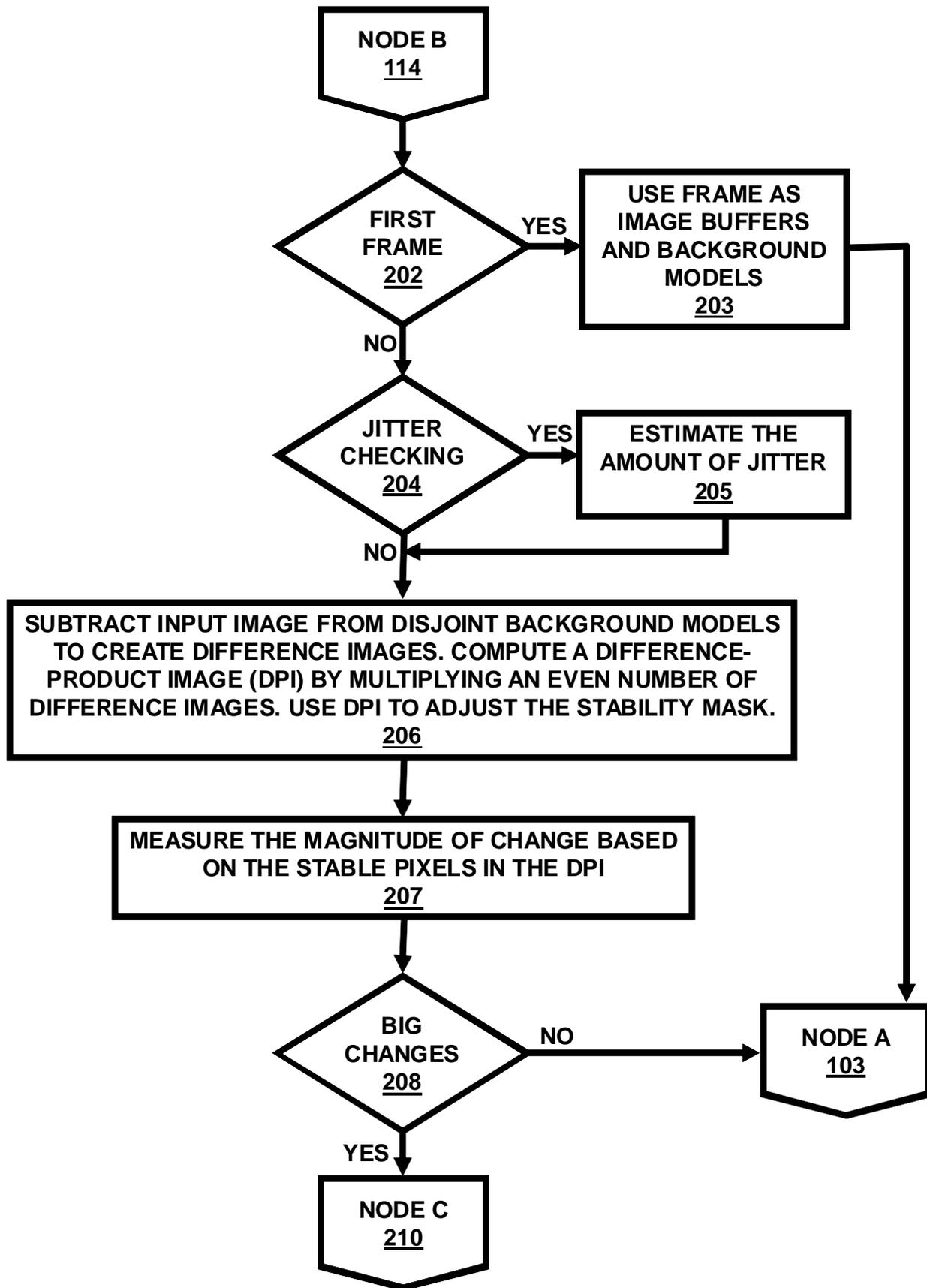
12. Kumar, R.; Sawhney, H.; Samarasekera, S.; Hsu, S.; Tao, H.; Guo, Y.; Hanna, K.; Pope, A.; Wildes, R.; Hirvonen, D.; Hansen, M.; Burt, P. Aerial Video Surveillance and Exploitation. *Proceedings of the IEEE*, 89 (10), pp. 1518–1539, Oct. 2001.
13. Bouttefroy, P.L.M.; Bouzerdoum, A.; Phung, S. L.; Beghdadi, A. , Vehicle Tracking using Projective Particle Filter. *Proc. Int. Conf. Advanced Video and Signal Based Surveillance*, pp. 7-12, Genoa, Italy, Sep. 2009.
14. He, Z.; Qin, Z.; Wen, H. Video-based Measure of Traffic Volume Parameter. *Proc. Int. Conf. on Automation and Logistics*, pp. 421–425, Jinan, China, Aug. 2007.
15. Bhargava, M.; Chen, C.; Ryoo, M. S.; Aggarwal, J. K. Detection of Abandoned Objects in Crowded Environments. *Proc. IEEE Conf. on Advanced Video and Signal Based Surveillance*, pp. 271–276, London, Sep. 2007.
16. Hu, W.; Tan, T.; Wang, L.; Maybank, S. A Survey on Visual Surveillance of Object Motion and Behaviors. *IEEE Trans. Systems, Man, and Cybernetics—Part C* **Aug. 2004**, 34 (3), 334–352.
17. Morris, B. T.; Trivedi, M. M. A Survey of Vision-Based Trajectory Learning and Analysis for Surveillance. *IEEE Trans. Circuits and Systems for Video Technology* **Aug. 2008**, 18 (8).
18. Lavee, G.; Rivlin, E.; Rudzsky, M. Understanding Video Events: A Survey of Methods for Automatic Interpretation of Semantic Occurrences in Video. *IEEE Trans. Systems, Man and Cybernetics—Part C* **Sep. 2009**, 39 (5), 489–504.
19. Candomo, J.; Shreve, M.; Goldgof, D. B.; Sapper, D. B.; Kasturi, R. Understanding Transit Scenes: A Survey on Human Behavior-Recognition Algorithms. *IEEE Trans. Intelligent Transportation Systems* **Mar. 2010**, 11 (1), 206–224.
20. Turaga, P.; Chellappa, R.; Subrahmanian, V. S.; Udrea, O. Machine Recognition of Human Activities: A Survey. *IEEE Trans. Circuits and Systems for Video Technology* **Nov. 2008**, 18 (1), 1473–1488.
21. Radke, R. J.; Andra, S.; Al-Kofahi, O.; Roysam, B. Image Change Detection Algorithms: A Systematic Survey. *IEEE Trans. Image Processing* **March 2005**, 14 (3), 294–307.
22. Tsai, D. M.; Lai, S. C. Independent Component Analysis-Based Background Subtraction for Indoor Surveillance. *IEEE Trans. Image Processing* **Jan. 2009**, 18 (1), 158–167.
23. Papanikolopoulos, N. P.; Khosla, P. K.; Kanade, T. Visual Tracking of a Moving Target by a Camera Mounted on a Robot: A Combination of Control and Vision. *IEEE Trans. Robotics and Automation* **Feb. 1993**, 9 (1), 14–35.

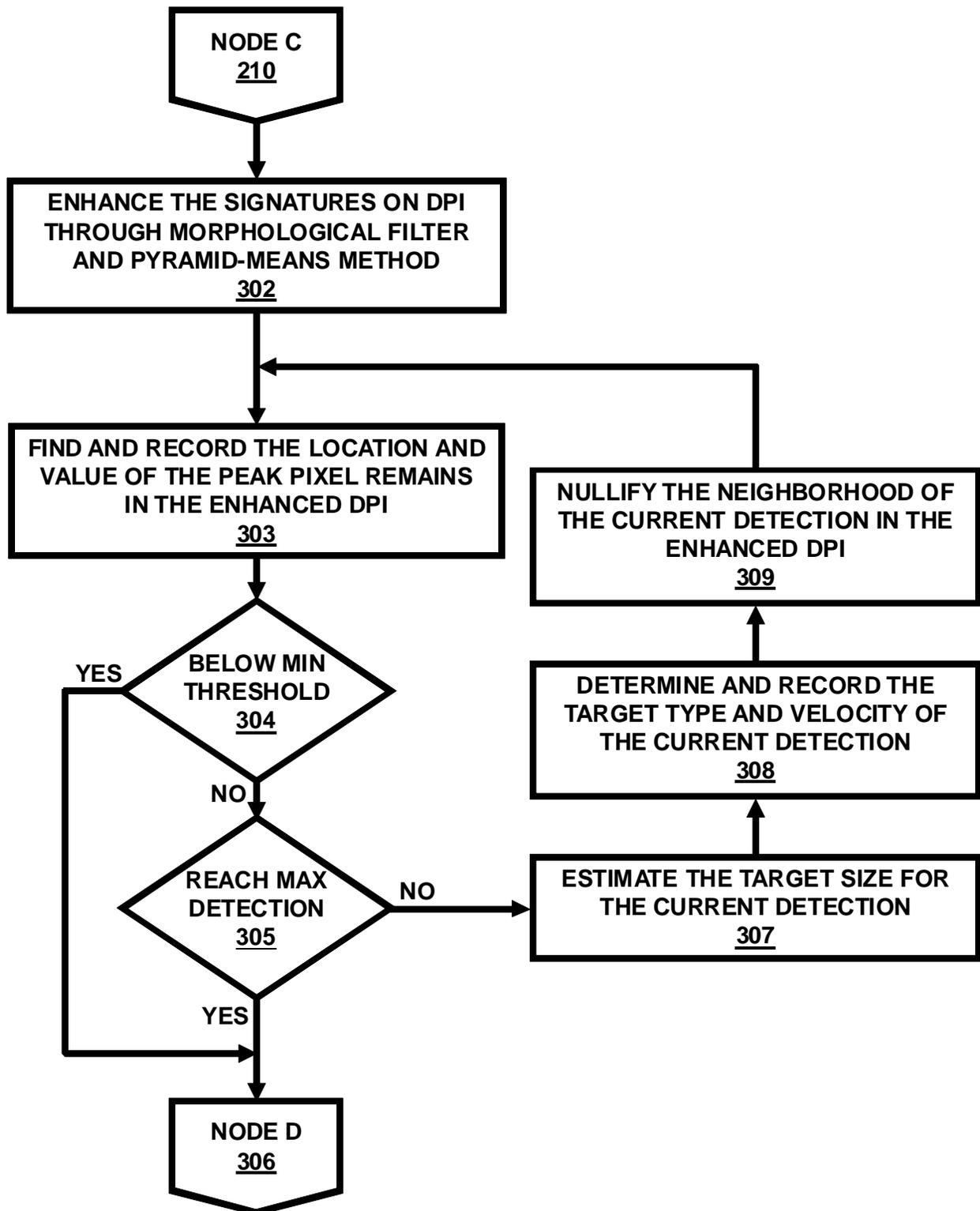
24. Chen, D.; Cannons, K.; Tyan, H.; Shih, S.; Liao, H. M. Spatiotemporal Motion Analysis for the Detection and Classification of Moving Targets. *IEEE Trans. Multimedia* **Dec. 2008**, *10* (8), 1578–1591.
25. Van Leeuwen, M. B.; Groen, F. C. Vehicle Detection with a Mobile Camera. *IEEE Robotics and Automation Magazine* **March 2005**, pp. 37–43.

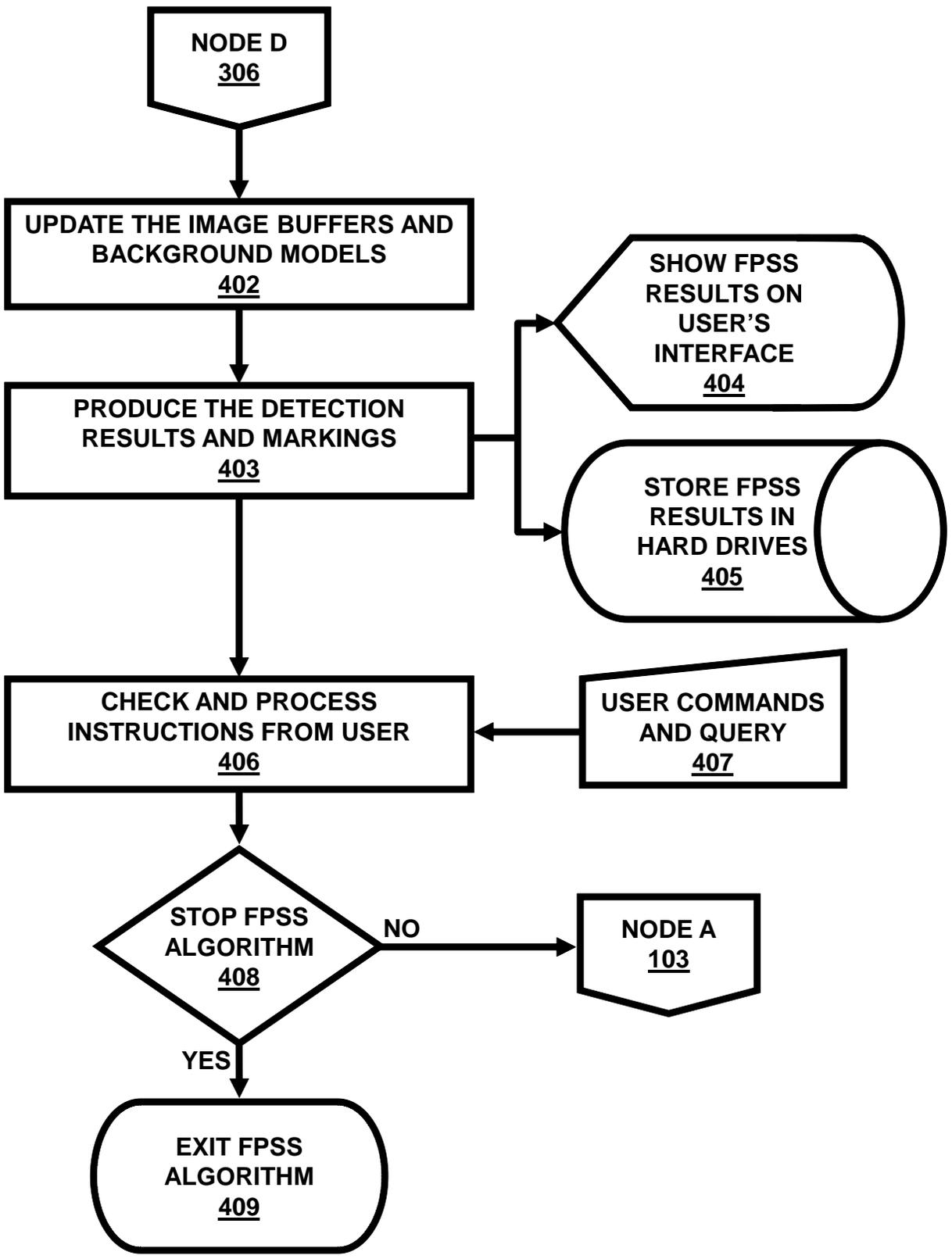
INTENTIONALLY LEFT BLANK.

Appendix A. Flowcharts for the FPSS Algorithm









List of Symbols, Abbreviations and Acronyms

AGC	Automatic gain control
ASCII	American Standard Code for Information Interchange
CCTV	Closed-circuit television
DPI	Difference-product image
DVR	Digital video recorder
FA	False alarm
FIFO	First-in first-out
FLIR	Forward-looking infrared
FOV	Field-of-view
FPA	Focal plane array
FPSS	Force protection surveillance system
GUI	Graphical user interface
IED	Improvised explosive device
LCD	Liquid crystal display
LWIR	Long-wave infrared
MTI	Moving target indication
NTSC	National Television Standards Committee
POD	Personnel observation device
ROC	Receiver operating characteristic

NO. OF COPIES	ORGANIZATION	NO. OF COPIES	ORGANIZATION
1 ELECT	ADMNSTR DEFNS TECHL INFO CTR ATTN DTIC OCP 8725 JOHN J KINGMAN RD STE 0944 FT BELVOIR VA 22060-6218	6	CECOM NVESD ATTN L. GRACEFFO ATTN M. GROENERT ATTN M. HERDLICK ATTN E EFKEMAN ATTN F PETITO ATTN D BRYSKI BLDG 305 10221 BURBECK RD FT BELVOIR VA 22060-5806
2	DARPA ATTN F PATTEN ATTN J RICKLIN 3701 N FAIRFAX DR ARLINGTON VA 22203-1714	2	CECOM NVESD ATTN C WALTERS ATTN J HILGER BLDG 307 10221 BURBECK RD FT BELVOIR VA 22060-5806
1	NGA ATTN R S RAND 12310 SURSISE VALLEY DR MAIL STOP DN 11 RESTON VA 20191-3449	1	US MILITARY ACDMY MATHEMATICAL SCI CTR OF EXCELLENCE ATTN MAJ J HARTKE PHOTONICS CENTER WEST POINT NY 10996-1786
1 CD	OFC OF THE SECY OF DEFNS ATTN ODDRE (R&AT) THE PENTAGON WASHINGTON DC 20301-3080	1	PM TIMS, PROFILER (MMS-P) AN/TMQ-52 ATTN B GRIFFIES BUILDING 563 FT MONMOUTH NJ 07703
1	US ARMY RSRCH DEV AND ENGRG CMND ARMAMENT RSRCH DEV & ENGRG CTR ARMAMENT ENGRG & TECHNLY CTR ATTN AMSRD AAR AEF T J MATTS BLDG 305 ABERDEEN PROVING GROUND MD 21005-5001	2	US ARMY ABERDEEN TEST CENTER ATTN TEDT AT WFT F CARLEN ATTN CSTE DTC AT TC N D L JENNINGS 400 COLLERAN ROAD ABERDEEN PROVING GROUND MD 21005-5059
1	US ARMY TRADOC BATTLE LAB INTEGRATION & TECHL DIRCTRT ATTN ATCH B 10 WHISTLER LANE FT MONROE VA 23651-5850		

NO. OF COPIES	ORGANIZATION	NO. OF COPIES	ORGANIZATION
1	US ARMY ERDC ATTN CEERD TR S 7701 TELEGRAPH RD BLDG 2592 ALEXANDRIA VA 22315	2	US ARMY RDECOM TARDEC ATTN AMSRD TAR R G R GERHART ATTN AMSRD TAR R J JASTER 6501 E ELEVEN RD MS 263 WARREN MI 48397-5000
1	US ARMY INFO SYS ENGRG CMND ATTN AMSEL IE TD A RIVERA FT HUACHUCA AZ 85613-5300	1	US ARMY SOLDIER & BIOLOGICAL CHEM CTR ATTN AMSSB RRT DP B LOEROP EDGEWOOD CHEM & BIOLOGICAL CTR BLDG E-5544 ABERDEEN PROVING GROUND MD 21010-5424
3	US ARMY MATERIEL SYS ANAL ACTVTY ATTN AMSRD AMS SC G KISTNER ATTN AMSRD AMS SC J MAZZ ATTN AMSRD AMS SC R WHEELER 392 HOPKINS RD ABERDEEN PROVING GROUND MD 21005-5071	1	COMMANDER USAISEC ATTN AMSEL TD BLAU BUILDING 61801 FT HUACHUCA AZ 85613-5300
1	US ARMY NATICK RDEC ACTING TECHL DIR ATTN SBCN TP P BRANDLER KANSAS STREET BLDG 78 NATICK MA 01760-5056	1	US GOVERNMENT PRINT OFF DEPOSITORY RECEIVING SECTION ATTN MAIL STOP IDAD J TATE 732 NORTH CAPITOL ST NW WASHINGTON DC 20402
1	US ARMY PM NV/RSTA ATTN SFAE IEW&S NV 10221 BURBECK RD FT BELVOIR VA 22060-5806	3	SITAC ATTN H STILES ATTN K WHITE ATTN R DOWNIE 11981 LEE JACKSON MEMORIAL HWY STE 500 FAIRFAX VA 22033-3309
3	COMMANDER US ARMY RDECOM ATTN AMSRD AMR J MILLS ATTN AMSRD AMR K DOBSON ATTN AMSRD AMR W C MCCORKLE 5400 FOWLER RD REDSTONE ARSENAL AL 35898-5000	1	US ARMY RSRCH LAB ATTN RDRL CIM G T LANDFRIED BLDG 4600 ABERDEEN PROVING GROUND MD 21005-5066
1	US ARMY RDECOM AMRDEC ATTN RDMR WS PL W DAVENPORT BLDG 7804 REDSTONE ARSENAL AL 35898		

NO. OF COPIES	ORGANIZATION
5	DIRECTOR US ARMY RSRCH LAB ATTN AMSRD ARL RO EL L DAI ATTN AMSRD ARL RO M D ARNEY ATTN AMSRD ARL RO MI R ZACHERY ATTN AMSRD ARL RO MM M-H CHANG ATTN RDRL ROI M J LAVERY PO BOX 12211 RESEARCH TRIANGLE PARK NC 27709-2211
59 HCS 1 CD	US ARMY RSRCH LAB ATTN IMNE ALC HRR MAIL & RECORDS MGMT ATTN RDRL CIM L TECHL LIB ATTN RDRL CIM P TECHL PUB ATTN RDRL D ATTN RDRL D J CHANG ATTN RDRL SE J PELLEGRINO ATTN RDRL SE J RATCHES ATTN RDRL SEE P GILLESPIE ATTN RDRL SES E D ROSARIO ATTN RDRL SES E G SAMPLES ATTN RDRL SES E H KWON ATTN RDRL SES E J DAMMANN ATTN RDRL SES E A CHAN (40 HCS, 1 CD) ATTN RDRL SES E M THIELKE ATTN RDRL SES E N NASRABADI ATTN RDRL SES E P RAUSS ATTN RDRL SES E R RAO ATTN RDRL SES E S HU ATTN RDRL SES E S YOUNG ATTN RDRL SES J EICKE ADELPHI MD 20783-1197
TOTAL:	104 (101 HCS, 2 CDS, 1 ELEC)