Distribution A: Approved for Public Release

Comparison of Relative Effectiveness of Video with Serial Visual Presentation for Target Reconnaissance from UASs

Frank E. Skirlo[a], Anthony J. Matthews[a], Melvin Friedman[b], Brian L. Mark[c]
[a]*CACI Technologies, Inc, Arlington, VA 22201;*
[b]*US Army RDECOM Communications-Electronics Research Development & Engineering Center, Night Vision and Electronic Sensors Directorate, Fort Belvoir, VA 22060;*
[c]*George Mason University, Department of Electrical and Computer Engineering, Fairfax, VA 22030*

ABSTRACT

Reconnaissance from an unmanned aerial systems (UAS) is often done using video presentation. An alternate method is Serial Visual Presentation (SVP). In SVP, a static image remains in view until replaced by a new image at a rate equivalent to the live video. Mardell et al. have shown, in a forested environment, that a higher fraction of targets (people lost in the forest), are found with SVP than with video presentation. Here Mardell's experiment is repeated for military targets in forested terrain at a fixed altitude. We too find a higher fraction of targets are found using SVP rather than video presentation. Typically it takes five seconds to cover a video field of view and at 30 frames per second. This implies that, for scenes where the target is not moving, 150 video images have nearly identical information (from a reconnaissance point of view) as a single SVP image. This is highly significant since transmission bandwidth is a limiting factor for most UASs. Finding targets in video or in SVP is an arduous task. For that reason we also compare aided target detection performance (Aided SVP) and unaided target detection performance on SVP images.

Keywords: Unmanned Aerial System, Reconnaissance, Aided Target Detection, Serial Visual Presentation, Target Detection

1.0 INTRODUCTION

1.1 Overview and Background. James Mardell (Department of Electrical and Electronic Engineering at Imperial College London),[1] conducted a noteworthy perception experiment that may have a major impact, if implemented, on the way the Department of Defense (DoD) conducts aerial reconnaissance using UASs. During his experiment, Mardell compared how many targets can be detected via video and SVP. His experiment used a simulated UAS for search and rescue (SAR) reconnaissance to locate lost individuals. To determine the time duration to capture sequential images, Mardell determined the length of the ground distance covered by the camera's field of view at the UAS's altitude. He then divided ground distance by the UASs airspeed. In Mardell's experiment, a UAS with an equivalent ground speed of 90 mph would have an equivalent SVP frame duration of 5.8 seconds. The results of Mardell's experiments are impressive. In the SVP mode, the observers detected 74.7% of all targets, while they detected 56.2% in the full motion video (FMV) mode. Appendix B has a table of acronyms**.**

1.2 Research Goals. Our goal is similar to Mardell's, with the exception of changing the experiment from finding people to finding tactical military targets in a forested background. We also wanted to determine if image processing can improve the ability of observers to detect targets in SVP images, i.e., conduct a third "Aided SVP" perception test. As with Mardell's experiment, our goal is to determine which presentation mode (FMV, SVP or aided SVP) detects the highest number of targets and the lowest number of false alarms. We are less interested in determining why humans can detect more (or less) targets in SVP or Aided SVP images versus FMV during UAS reconnaissance missions. SVP or Aided SVP has tremendous potential to save Ku SATCOM bandwidth and has significant potential to detect targets at a greater rate than FMV. The anticipated reduced SATCOM bandwidth requirement from a single UAS by using SVP or aided SVP, allows for an increased numbers of UAS reconnaissance missions within a theater of operations by sharing SATCOM resources (applying multiple access techniques to send images over a SATCOM channel).

Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXVII,
edited by Gerald C. Holst, Keith A. Krapels, Proc. of SPIE Vol. 9820, 982008
2016 SPIE · CCC code: 0277-786X/16/$18 · doi: 10.1117/12.2224796

1.3 Approach. We followed Mardell's approach using video and imagery from a UAS simulation system called Multiple Unified Simulation Environment (MUSE)/Air Force Synthetic Environment for Reconnaissance and Surveillance (AFSERS)[2] at the U.S. Army Night Vision and Electronic Sensors Directorate (NVESD). MUSE/AFSERS has the ability select from a variety of U.S. UASs, and apply a wide variety of Electro-Optic/Infrared (EO/IR) cameras. MUSE/AFSERS can fly the UAV in any search pattern and altitude to view fixed or moving targets, or cover a pre-established flight pattern and conduct a route or area reconnaissance while keeping its camera fixed at a specific heading and down angle. MUSE/AFSERS also has functionality to capture and store fixed frame imagery (FFI) used for SVP. Both the captured video and images were migrated onto a computer where individual observers can search for targets during three perception tests: FMV, SVP, and Aided SVP. For the Aided SVP Test, MATLAB®[3] Image Processing Toolbox was used to help highlight and accentuate potential targets within the image by placing bounding boxes around them.

1.4 Results. We conducted initial FMV, SVP, and Aided SVP perception experiments for forested terrain only, though our desire was to also conduct experiments using desert and urban environments. We limited the FMV reconnaissance mission to about 30 minutes, which equated to 309 images used for the SVP and Aided SVP tests. We only had time to have four observers undergo the three perception tests. Though limited, results showed observers generally detected the most target with Aided SVP, followed by SVP, followed by FMV.

1.5 Conclusions. Though we conducted a limited test, our results are similar to Mardell's results—i.e., detection of targets using SVP yielded better target acquisition performance than video. Additionally, Aided SVP performed better than SVP. In the future, we will need to increase the time for simulated missions, increase the number of targets, and include desert and urban terrains. There are many image processing algorithms that filter pixel and connected sub-images to detect potential targets in various backgrounds. Additional algorithms should be explored to maximize accuracy of finding potential targets while minimizing false ones. Once we conduct more and enhanced simulation tests, additional live UAS FMV, SVP, and Aided SVP experiments should be conducted to confirm our simulated results.

2.0 EXPERIMENTATION METHOD

2.1 Overview. Conducting reconnaissance from UASs using real-time video may not be the most efficient use of one of the most critical and limited resources within a theater of operations--satellite communications (SATCOM) bandwidth. While UASs conduct FMV reconnaissance of a swath of the surface of an area of interest, the UAS transmits the video, at a rate of 30 frames a second, typically through a Ku-band commercial SATCOM satellite to a ground station, which receives the video feed for analysis. Usually video feeds are evaluated, in real-time, at a Processing, Exploitation, and Dissemination (PED) Center for larger strategic UAV assets, or a Tactical Operations Center (TOC) for smaller, tactical UAS assets. PEDs can be located in or outside the theater of operations. The below Operational View-1 (OV-1) in Figure 1 depicts the various systems involved:

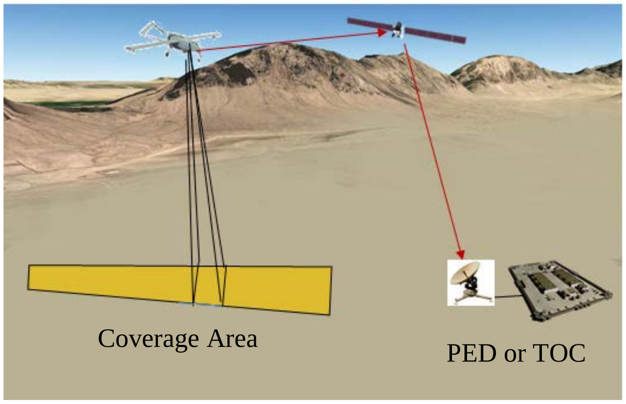


Figure 1. OV-1 of UAV collecting video of area of interest and transmitting it back to a Processing, Exploitation, and Dissemination (PED) Center or Tactical Operations Center (TOC).

2.2 FMV, SVP, and Aided SVP Imagery. We flew the simulated UAS in a near straight-line flight path, recording both video of the EO camera feed while taking periodic images over the same coverage area. Target locations were established as designated way points to ensure that all targets were captured by both video and fixed frame imagery. Figure 2 shows the three modes the UAS used to capture FMV, and SVP and Aided SVP images.

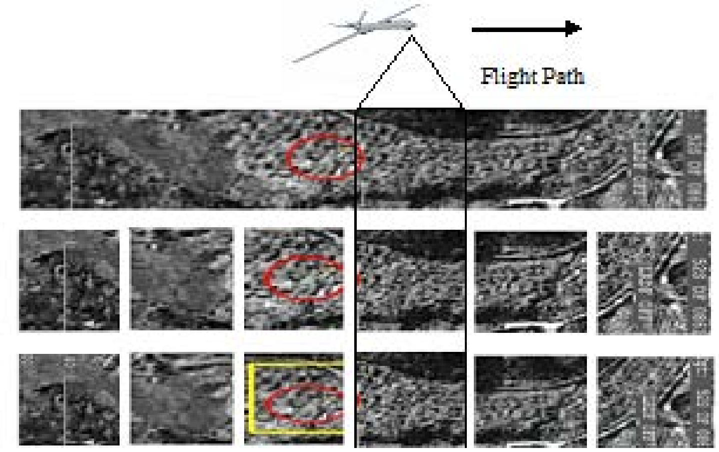


Figure 2. FMV mode (top row), SVP mode (middle row), and Aided SVP mode (bottom row); targets circled in red[1]

2.3 Perception Test Set-up. As illustrated in Figure 3, for all three perception tests we used a simulated U.S. Army Shadow 400 UAS. We applied a low resolution electro-optic (EO) camera (provided by Night Vision Image Generator)[4] and flew the aircraft approximately 2500 feet above ground level (AGL), at an airspeed (v) of 70 knots (36.01 m/s). The camera was pointed at a fixed -60 degree elevation at a 0 degree azimuth. We initially desired to conduct these tests using varying altitudes, but determined that lower altitudes had too narrow a field of view (FOV) for its EO sensor and target view time was too fast to adequately detect targets. Flying the UAS at too high an altitude made targets far too challenging to detect for all three perception tests. The MUSE/AFSERS simulator allows one to observe real-time video, and capture periodic SVP images and store them in a target file. For the FMV test, we flew the UAS in a straight-line reconnaissance mission for twenty minutes, long enough to start causing observer fatigue. Figure 4 shows representative true imagery. For the SVP and Aided SVP tests, we set the frequency of the SVP image captures based on the UAS's airspeed. The length of the ground coverage zone for the sensor (d) was approximately 180 meters, therefore the rate was calculated as t = 5.1 seconds (t = d/v). Altitudes varied due to varying terrain changes and simulated UAS aerodynamics.

Figure 3. RQ-7B Shadow Unmanned Aerial Vehicle

2.4 Target Insertion. We inserted various weapons systems and equipment using models provided by Vignette Planning and Rehearsal Software (ViPRS)[5]. Simulated targets inserted into the environment were chosen from those represented in Table 1. For our simulation, we inserted a total of ten targets randomly throughout the flight path in locations that were feasible in a tactical scenario.

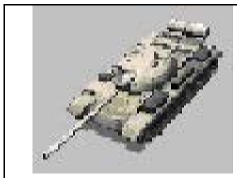
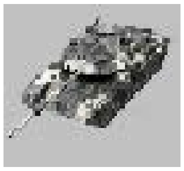
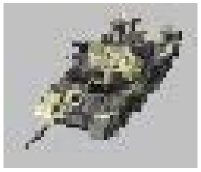
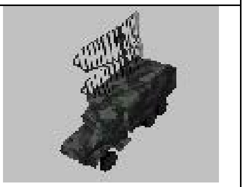
					
T-55	**T-72**	**T-80**	**BMP1**	**BMP2**	**BRDM2**
					
SCUD Desert	**GAMMON**	**ZSU 23-4**	**DOG EAR**	**MLRS Gen Desert**	**FLAT FACE**

Table 1: ViPRS Targets used in Perception Tests

2.5 Test Procedure. VideoLAN Client® (VLC) Media Player[6] was used to record video from the EO camera at the same time periodic SVP images were captured. MATLAB programs were created to run video recordings, SVP images, and Aided SVP images, enabling observers undergoing the perception testing the ability to left-click a detected target (these could include both true and false detections), place a red circle around the selected target, and place it in a grade file as jpeg images. Figure 4 shows representatives images selected with targets in the SVP mode.

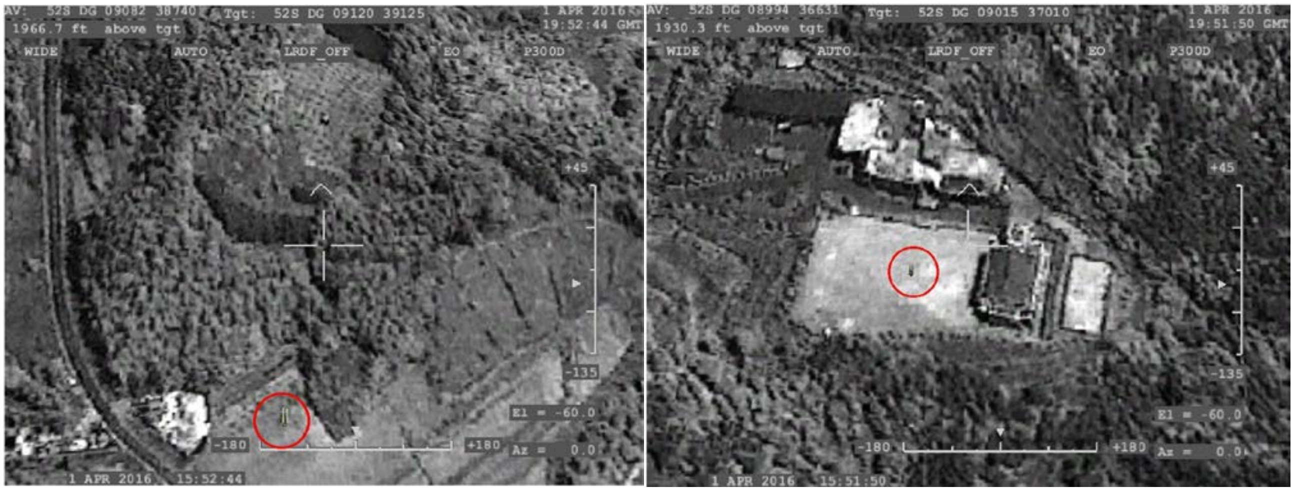


Figure 4: Targets selected using MATLAB programs during perception tests

For the Aided SVP Test, MATLAB Image Processing Toolbox was used to highlight and accentuate potential targets within the image by placing bounding boxes around them. We applied image segmentation to separate targets from their backgrounds using image processing by color, shape (morphological), and size. Some of the key MATLAB commands that helped accomplish these tasks will now be described. Many of the toolbox features were obtained by watching the Mathworks “Image Processing Made Easy” Webinar.[7] Figure 5 below depicts a typical ViPRS target within a forested background that we wish to process (at tip of yellow arrow).

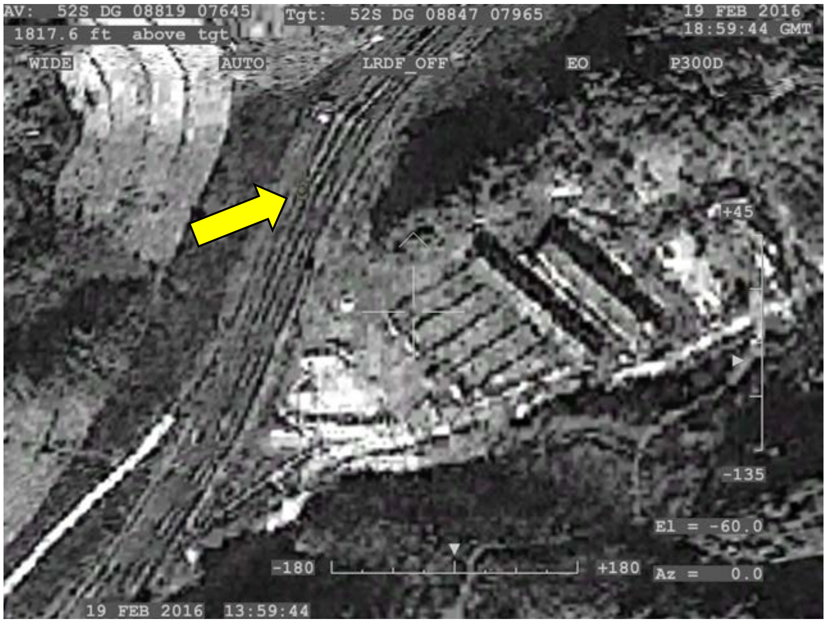


Figure 5. Sample SVP image with targets

2.6 Aided SVP Image Processing.

Pre-Processing. The above jpeg image was read into a MATLAB program using the `imread` command (see Appendix A). We then increased the contrast of the image within each of the Red-Green-Blue (RGB) planes using the using the `imadjust` command (See Figure 6.b). This command reduces RGB values of 0.3 to 0, and increases the respective RGB values of 0.8/0.9/0.8 to 1, producing the second image below. We then removed noise from and sharpened the image using the `imfilter` command (see Figure 6.c). This command convolves the image with a 3x3 ones matrix (normalized by multiplying by 1/9) which acts as a low-pass filter. The third image below is the filtered image. The image may not aid the observer in identifying a target against the scene, but is useful for the adided target algorithm.

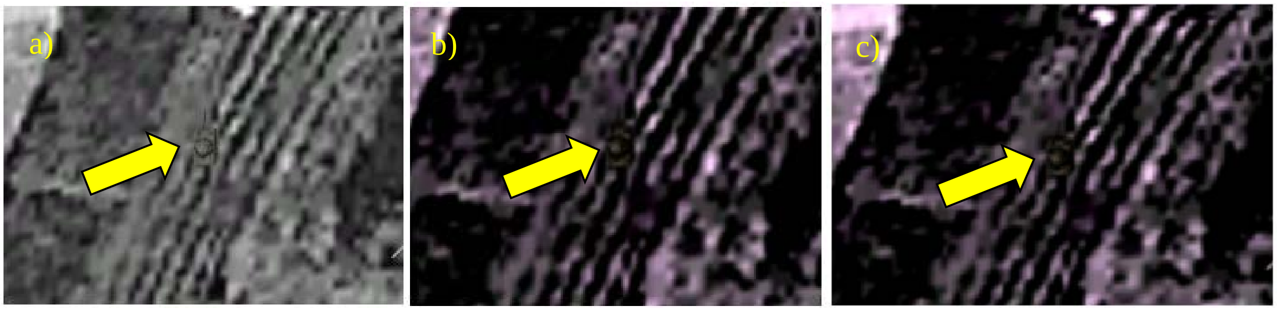


Figure 6. a) Magnified SVP image from Figure 5, b) After increasing contrast using `imadjust` command, c) After noise filtering using `imfilter` command

Target Segmentation. We then detect targets by evaluating the RGB pixel values. When studying target and background pixel values, we note that red and blue pixel values are almost identical for target pixels, and that red pixels have greater values then blue ones for background pixels. As an example, a background RGB pixel can be [18 19 11] (to the right of the tank); one of the RGB pixel values for the T-55 tank is [72 74 71]. Therefore, we developed an algorithm that identified blue-plane pixels that are greater than or equal to 85% of the values of the red-plane pixels. If this condition was met, the pixel value was changed to black ([0 0 0]), otherwise it was changed to white ([256 256 256]), thereby creating a binary image. This bit map now displays the connected image of a white tank on a black background) after the program conducted a compliment of the bit map. Figure 7 shows the original and magnified portion of the image containing the target.

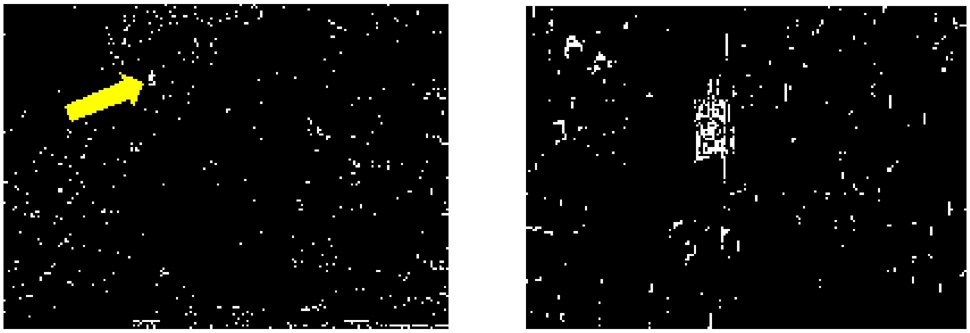


Figure 7. Original and magnified target bit map

We add the red and green planes of the above bit map to increase the target's resolution. Figure 8 is the resultant image. The `bwconncomp` command finds and stores the connected components within the binary image.

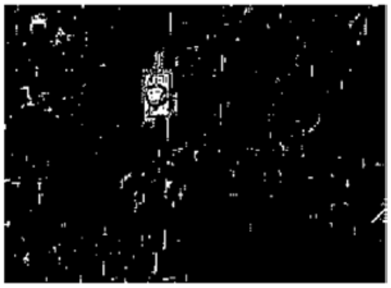


Figure 8. Red plane added to Green Plane on target bit map

Processing Connected Components in Image. Now that we have identified the connected components, we desire to place bounding boxes around the targets on the original images. We apply the `regionprops` command to find various properties of the connected shapes in CC.[8] In our case, the command returns a data array called "stats" with elements corresponding to each of the elements in "labeled." In each element in "stats, there are four values or arrays regarding each connected component within the image. "Eccentricity" refers to conic section properties associated with the connected component (a "0" value is a perfect circle, a value of "1" is a line segment, a value between 0 and 1 is an ellipse), "Area" is the computed area of the connected component, "BoundingBox" provides a 1x4 array with the upper left coordinates of the bounding box that would overlay around the target along with the x- and y-axis lengths of the box, and "Centroid" provided the center (x,y) location of the connected component. The array "areas" lists the area associated for each of the connected components.

Filtering Components by Size. We now filter out the connected components by size, setting minimum and maximum areas of the potential targets using the following MATLAB code. If the areas of the connected components do not meet the size threshold of 70 < area(M) < 350, they are defaulted to zero.

We use the command `find(area)`to find the array element number that contained areas that met the above thresholds, and stores it into array "numPlane". The command "stats" stores the four `regionprops` values (Eccentricity, Area, BoundingBox, and Centroid) into the data array statsDefects.

Placing Bounding Boxes around Potential Targets. At this point, we are ready to place bounding boxes around potential targets onto the original background using the following code. We display the original unprocessed image I and find the centroids of each of the connected components having areas with the predetermined thresholds. We start the bounding boxes over the centroids. Bounding boxes are adjusted to ensure they encompass the target. If either of the side lengths are 50 or greater, then the box is not displayed (guaranteed false positive). We use the color yellow for the bounding boxes using the RGB color code [1 1 0] at a width of 2.5. Two of the bounding boxes depict false positives (extreme left and lower right). Lastly, the program counts and displays the total number of possible target on a title banner above the images. Figure 9 shows an overall and magnified view of the original scene (I) overlaid with bounding boxes around the possible targets. Notice that there are bounding boxes around two false targets to the right of the tank.

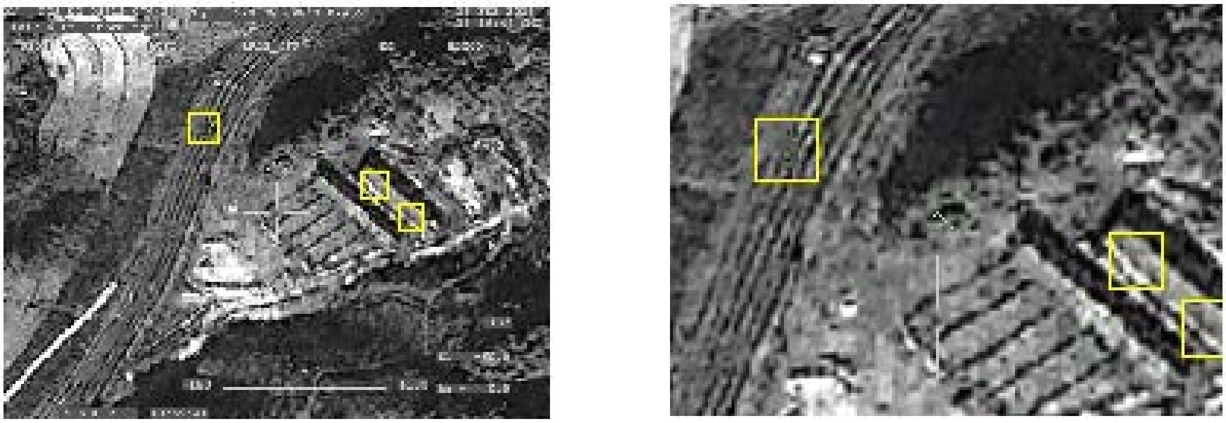


Figure 9. Aided SVP images, original and magnified

3.0 RESULTS

Due to lack of time, we were only able to have four observers undergo all the three perception tests (FMV, SVP, and Aided SVP). We identified the observers as A-D. All were staff assigned to NVESD (three were Department of the Army civilians; one was a contractor). All were technically oriented (STEM degrees) and had backgrounds in M&S systems; two had experience with perception testing. Table 2 provides a score of how many targets were successfully detected for each test. There were a total of ten targets within the video and SVP images.

	FMV		SVP		Aided SVP	
Observer	TP	FP	TP	FP	TP	FP
A	4	5	7	0	9	0
B	3	1	4	1	8	0
C	8	12	7	0	10	0
D	6	5	8	1	9	0

Table 2: Results for three perception tests; total targets detected (FP = false positives, TP = true positives), N = 10 total targets

With SVP, Observers A, B, and D were able to detect more targets (TPs) and had fewer FPs than with FMV. Observer C had the one fewer TP detection, but had drastically fewer FPs. All observers detected 9 or 10 of the targets with no FPs with the Aided SVP test. We had other observers attempt to take the FMV and SVP tests, but had difficulty with them for several reasons (e.g., over-anticipated each image having targets, software locked up during test, or failure to understand the purpose of the test). We omitted these results, but will ensure these issues will be addressed in future tests.

I provided an e-mail questionnaire to all the participants and asked them to rate the value of SVP in target detection:

1. Target detection was easier using SVP over FMV (rate 1-5, 1 = strongly disagree, 3 = neutral, 5 = strongly agree)
2. Target detection was easier using Aided SVP over SVP (same rating criteria as above)
3. Provide any recommendations to improve any of the three perception tests (see Appendix C for responses)

Table 3 is a summary of the responses provided by the observers on statements 1 and 2:

Observer	Response 1	Response 2
A	5	5
B	5	5
C	4	4
D	4	5
Average	4.5	4.75

Table 3: Results from Questionnaire Statements 1 and 2

4.0 CONCLUSIONS

Though we conducted a limited test, our results are similar to Mardell's—we achieved a higher detection of targets using SVP than with video, and a lower FP rate with SVP than with video. Additionally, Aided SVP performed better than SVP. In future experiments, we will need to increase the observers, time for simulated missions, and the number of targets, and include desert and urban terrains to confirm our initial results. On average, observer questionnaire responses indicate that target detection using SVP was easier than in FMV mode, and that Aided SVP was easier than SVP. We could apply some of the recommendations provided by the observers such as to provide additional training and practice experiments for observers.

There are many image processing algorithms that filter pixel and connected sub-images for certain targets within their backgrounds. These would need to be changed or adjusted for the types of targets one is trying to detect within the background where they are located. We used one algorithm, but others should be studied to find which ones maximized TPs (actual targets) while minimizing FPs.

Once we confirm our results with enhanced simulation tests, we can study how to more efficiently compress Aided SVP images in order to minimize SATCOM bandwidth requirements. Additionally, we can determine if alternate SATCOM multiple access techniques such as TMDA, ALOHA, or Slotted ALOHA could be used to transmit images over a single SATCOM channel for analysis, exploitation, and processing.[9]

REFERENCES

[1] Mardell, J., Witkowski, M., and Spence, R., "A Comparison of Image Modes for a Visual Search and Rescue Task," Behaviour & Information Technology. Sep 2014, Vol. 33: 905-91

[2] MUSE/AFSERS Software, developed by the Joint Technology Lab Systems Integration Lab (SIL, Redstone Arsenal, AL), Version 9.0.4.0

[3] MATLAB Software developed by Mathworks, version R2013a

[4] Night Vision Image Generator (NVIG) Software, Version 15.2

[5] Vignette Planning and Rehearsal Software (ViPRS), Version 3.1.3.0

[6] VLC Media Player Software, Version 2.1.3

[7] The', A., "Image Processing Made Easy," 5 September 2014, https://www.youtube.com/watch?v=1-jURfDzP1s (28 March 2016)

[8] Mathwork Help Site, http://www.mathworks.com/help/matlab/ref/help.html (5 April 2016)

[9] Leon-Garcia, A. and Widjaja, I., [Communications Networks], Tata McGraw-Hill Publishers, New Delhi & New York, 377-380, 400-401 (2004)

APPENDIX A – Image Processing Algorithm for Aided SVP

```
I = imread('image136.jpg');

RGB = imadjust(I,[.3 .3 .3; .8 .9 .8],[]);
RGB = imcrop(RGB,[110 110 1170 850]);
h = ones(3,3)/9;

RGB2 = imfilter(RGB,h);

for col = 1:1170
    for row = 1:850

        if RGB2(row,col,3) >= 0.85* RGB(row,col,1)
            RGB2(row,col,:) = 0;
        else
            RGB2(row,col,:) = 256;
        end

    end

end

clear Array;

RGB2 = imcrop(RGB2,[2 2, 1068 775]);

CC = bwconncomp(RGB2(:,:,1) + RGB2(:,:,2));
stats = regionprops(CC,'Eccentricity','Area','BoundingBox');
areas = [stats.Area];

n = length(areas);
for m = 1:n
    if areas(m) < 70   % minimum Target Box area
        areas(m) = 0;
    end
    if areas(m) > 350   % Maximum Target Box area
        areas(m) = 0;
    end
end

numPlane = find(areas);
statsDefects = stats(numPlane);

figure(1),imshow(I); %title(sprintf('Figure %d', Inum),'FontSize',16);


%Sets figure size to whole screen
set(gcf,'units','normalized','outerposition',[0 0 1 1])

%Forces image to be displayed in figure now
drawnow

%Adds bbox to image unless its width/height > 50

for idx = 1:length(numPlane)
    Box = statsDefects(idx).BoundingBox;
    if (Box(3) >= 50 || Box(4) >= 50) || (Box(4) < Box(3))
        continue
    end
    h2 = rectangle('Position',[Box(1)+85,Box(2)+85,Box(3)+50,Box(4)+50]);
    set(h2,'EdgeColor',[1  1  0],'Linewidth',2),     % Red = 0.75,0,0
    hold on;
end

%Adds that number of targets as a title
count = nnz(areas);
title([Iname, ' Possible target count: ', num2str(count)]);
```

APPENDIX B – Acronyms

AFSERS	Air Force Synthetic Environment for Reconnaissance and Surveillance
AGL	Above Ground Level
ALOHA	Application Layer Optimization and High Availability
DoD	Department of Defense
EO	Electro-Optic
EO/IR	Electro-Optic/Infrared
FFI	Fixed Frame Imagery
FMV	Full Motion Video
FOV	Field of View
FP	False Positive
LAN	Local Area Network
MATLAB	Matrix Laboratory
M&S	Modeling and Simulation
MUSE	Multiple Unified Simulation Environment
NVESD	Night Vision and Electronic Sensors Directorate
OV	Operational View
PED	Processing, Exploitation, and Dissemination
Pixel	Picture Element
RDECOM	Research, Development, and Engineering Command
RGB	Red-Green-Blue
SAR	Search and Rescue
SATCOM	Satellite Communications
SVP	Serial Visual Presentation
STEM	Science, Technology, Engineering, and Mathematics
TMDA	Time Division Multiple Access
TOC	Tactical Operations Center
TP	True Positive
UAS	Unmanned Aerial System
VLC	Video LAN Client
ViPRS	Vignette Planning and Rehearsal Software

APPENDIX C – Responses to Questionnaire on How to Improve Perception Tests

- On a scale from 1-10 (1 lowest, 10 highest), I would rate FMV at a 2 for detecting targets, SVP at a 4, and Aided SVP at an 8
- A short pre-test should be given prior to test to provide observers an idea of what they are looking for
- Observer should be able to discard FP detection during FMV test
- Need a more comprehensive in-brief on what targets should look like
- Let observers know up front this is an EO, not an IR test
- Bounding boxes placed on targets helped provide confirmation that you were seeing a valid target
- Taking the successive tests may help your ability to detect targets