Scientific Research Publishing

Deep Learning Based Target Tracking and Classification for Infrared Videos Using Compressive Measurements

Chiman Kwan[1*], Bryan Chou[1], Jonathan Yang[2], Trac Tran[3]

[1]Applied Research LLC, Rockville, MD, USA
[2]Google, Inc., Mountain View, CA, USA
[3]Department of Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD, USA
Email: *chiman.kwan@signalpro.net

Abstract

Although compressive measurements save data storage and bandwidth usage, they are difficult to be used directly for target tracking and classification without pixel reconstruction. This is because the Gaussian random matrix destroys the target location information in the original video frames. This paper summarizes our research effort on target tracking and classification directly in the compressive measurement domain. We focus on one particular type of compressive measurement using pixel subsampling. That is, original pixels in video frames are randomly subsampled. Even in such a special compressive sensing setting, conventional trackers do not work in a satisfactory manner. We propose a deep learning approach that integrates YOLO (You Only Look Once) and ResNet (residual network) for multiple target tracking and classification. YOLO is used for multiple target tracking and ResNet is for target classification. Extensive experiments using short wave infrared (SWIR), mid-wave infrared (MWIR), and long-wave infrared (LWIR) videos demonstrated the efficacy of the proposed approach even though the training data are very scarce.

Keywords

Target Tracking, Classification, Compressive Sensing, SWIR, MWIR, LWIR, YOLO, ResNet, Infrared Videos

1. Introduction

There are many applications such as traffic monitoring, surveillance, and security monitoring that use optical and infrared videos [1]-[6]. Object features in

optical and infrared videos can be clearly seen as compared to radar based trackers [7] [8].

Compressive measurements [9] [10] are normally collected by multiplying the original vectorized image with a Gaussian random matrix. Each measurement contains a scalar value and the measurement is repeated M times where M is much fewer than N (the number of pixels). To track a target using compressive measurements, it is normally done by reconstructing the image scene and then conventional trackers are then applied. There are two drawbacks in this conventional approach. First, the reconstruction process using L_0 [11] or L_1 [12] [13] [14] based methods is time consuming, which makes real-time tracking and classification impossible. Second, there may be information loss in the reconstruction process [15].

In the literature, there are some trackers such as [23] that use the term compressive tracking. However, those trackers are not using compressive measurements directly. There are several advantages if one can directly perform target tracking and classification using compressive measurements. First, because reconstruction of video frames from compressive measurements using Orthogonal Matching Pursuit (OMP) or Augmented Lagrangian Method with L1 (ALM-L1) are time consuming, direct tracking and classification in compressive measurement domain will enable near real-time processing. Second, it is well-known that reconstruction tends to lose information [15]. Working directly using compressive measurement will generate more accurate tracking and classification results [15]-[22].

Recently, we developed a residual network (ResNet) [24] based tracking and classification framework using compressive measurements [10]. The compressive measurements are obtained by using pixel subsampling, which can be considered as a special case of compressive sensing. ResNet was used in both target detection and classification. The tracking is done by detection. Although the performance in [10] is much better than conventional trackers, there is still room for further improvement. The key area is to improve the tracking part, which has a significant impact on the classification performance. That is, if the target area is not correctly located, the classification performance will degrade.

In this paper, we propose an alternative approach, which aims to improve the tracking performance. The idea is to deploy a high performance tracker known as YOLO [25] for target tracking. YOLO is fast, accurate, and has comparable performance as other trackers such as Faster R-CNN [26]. It should be noted that YOLO is used for object detection and not for object tracking. The YOLO for tracking is done by object detection. That is, we custom train YOLO for detecting certain vehicles and the detection results (target location information) from each frame are recorded and then tracked. This is known as tracking by detection. The detection results (bounding boxes of objects) are fed into a classifier. The classification is using ResNet because ResNet has better classification than the default classifier in YOLO.

It is emphasized that a preliminary version of this paper was presented in an SPIE conference [27] in which we only focused on SWIR videos. Here, we have significantly expanded the earlier paper to include additional experiments using MWIR, and LWIR videos. The experiments clearly demonstrated that the performance of the proposed approach is accurate and applicable to different types of infrared videos. Moreover, another contribution of this paper is that our study is the first comprehensive study of vehicle tracking and classification of several types of infrared videos directly in compressive measurement domain (subsampling).

This paper is organized as follows. Section 2 describes the idea of compressive sensing via subsampling, YOLO detector, and ResNet. Section 3 presents the tracking and classification results directly in the compressive measurement domain using SWIR videos. Section 4 focuses on tracking and classification of vehicles in MWIR videos. Section 5 repeats the studies for LWIR videos. In all cases, a comparative study of YOLO and ResNet for classification is also presented. Finally, some concluding remarks and future research directions are included in Section 6.

2. Background

2.1. Compressive Sensing via Subsampling

Using Gaussian random to generate compressive measurement makes the target tracking very difficult. This is because the targets can be anywhere in a frame and the target location information is lost in the compressive measurements. To resolve the above issue, we propose a new approach in which, instead of using a Gaussian random sensing matrix, we use a random subsampling operator (*i.e.*, keeping only a certain percentage of pixels at random from the original data) to perform compressive sensing. This is similar to using a sensing matrix by randomly zeroing out certain elements from the diagonal of an identity matrix. Figure 1 displays two examples of a random subsampling sensing matrices. Figure 1 shows a subsampling operator which randomly selects 50% of the pixels in a vectorized image. Figure 1(b) shows the equivalent case of randomly selecting 50% of the pixels in a 2-D image.

(a)   (b)
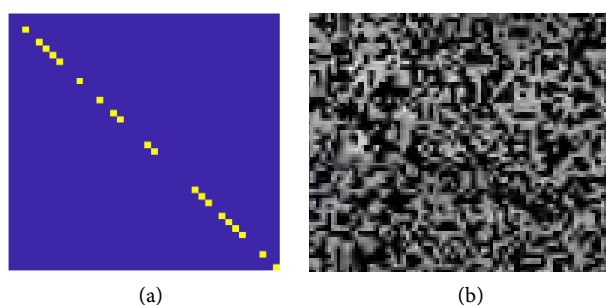
Figure 1. (a) Visualization of the sensing matrix for a random subsampling operator with a compression factor of 2. The subsampling operator is applied to a vectorized image. This is equivalent to applying a random mask shown in (b) to an image.

2.2. YOLO

We used the so-called tracking by detection approach. In the target tracking literature, there are several ways to carry out tracking. Some trackers such as STAPLE [28] or GMM [29] require an operator to put a bounding box on a specific target and then the trackers will try to track this initial target in subsequent frames. The limitation of this type of trackers is that they can track one target at a time. Another limitation is that they cannot track multiple targets simultaneously. Other trackers such as YOLO and Faster R-CNN do not require initial bounding boxes and can simultaneously detect objects. We can call the second type of trackers: tracking by detection. That is, based on detection results, we determine the vehicle locations in all the frames.

YOLO tracker [25] is fast and has similar performance as Faster R-CNN [26]. We picked YOLO because it is easy to install and is also compatible with our hardware, which seems to have a hard time to install and run Faster R-CNN. The input image is resized to 448×448. There are 24 convolutional layers and 2 fully connected layers. The output is $7 \times 7 \times 30$. We have used YOLOv2 because it is more accurate than YOLO version 1. The training of YOLO is quite simple. Images with ground truth target locations are needed. The bounding box for each vehicle was manually determined using tools in MATLAB. For YOLO, the last layer of the deep learning model was re-trained. We did not change any of the activation functions. YOLO took approximately 2000 epochs to train.

YOLO also comes with a built-in classification module. However, based on our evaluations, the classification accuracy using YOLO is not good as can be seen in Sections 3 - 5. This is perhaps due to a lack of training data.

2.3. ResNet Classifier

The ResNet-18 model is an 18-layer convolutional neural network (CNN) that has the advantage of avoiding performance saturation and/or degradation when training deeper layers, which is a common problem among other CNN architectures. The ResNet-18 model avoids the performance saturation by implementing an identity shortcut connection, which skips one or more layers and learns the residual mapping of the layer rather than the original mapping.

Training of ResNet requires target patches. The targets are cropped from training videos. Mirror images are then created. We then perform data augmentation using scaling (larger and smaller), rotation (every 45 degrees), and illumination (brighter and dimmer) to create more training data. For each cropped target, we are able to create a data set with 64 more images.

3. Tracking and Classification Results Using SWIR Videos

Our research objective is to perform tracking and classification of three trucks using the sponsor provided SWIR videos. One video (Video 4) starts with vehicles (Ram, Frontier, and Silverado) leaving a parking lot and moves on to a remote location. Another video (Video 5) is just the opposite. These videos are

challenging for several reasons. First, the target sizes vary a lot from near field to far field. Second, the target orientations also change drastically from top view to side view. Third, the illuminations in different videos are also different. Here, the compressive measurements are collected via direct sub-sampling. That is, 50% or 75% of the pixels are thrown away during the data collection process.

In our earlier paper [10], we have included some tracking results where conventional trackers such as GMM [29] and STAPLE [28] were used. The tracking performance was poor when there are missing data.

3.1. Tracking Results

We experimented with a YOLO tracker, which has been determined to perform better tracking than our earlier ResNet based tracker [10]. We used the following metrics for evaluating the tracker performance:

- Center Location Error (CLE): It is the error between the center of the bounding box and the ground-truth bounding box.
- Distance Precision (DP): It is the percentage of frames where the centroids of detected bounding boxes are within 20 pixels of the centroid of ground-truth bounding boxes.
- EinGT: It is the percentage of the frames where the centroids of the detected bounding boxes are inside the ground-truth bounding boxes.
- Number of frames with detection: This is the total number of frames that have detection.

Conventional Tracker Results

We applied the GMM tracker to one of our videos. From the results shown in Figure 2, it can be seen that the tracking results are not satisfactory even when there are no missing pixels. In some frames, the GMM tracker simply lost the targets.

STAPLE [28] is one of the high performing trackers in recent years. For this algorithm, the histogram of oriented gradients (HOG) features are extracted from the most recent estimated target location and used to update the models of the tracker. Then a template response is calculated using the updated models and the extracted features from the next frame. To be able to estimate the location of the target, the histogram response is needed along with the template response. The histogram response is calculated by updating the weights in the current frame. Then the per-pixel score is computed using the next frame. This score and the weights, calculated before, are used to determine the integral image, and ultimately, the histogram response. Together, with the template and histogram response, the tracker is able to estimate the location of the target.

Figure 3 shows good tracking results when there are no missing data. The green boxes show the target locations. However, when 50% of the pixels are missing, the tracking performance deteriorates significantly as shown in Figure 4.

Tracking Results: Train using Video 4 and Test using Video 5

We have two SWIR videos from the AF. Here, we used Video 4 for training
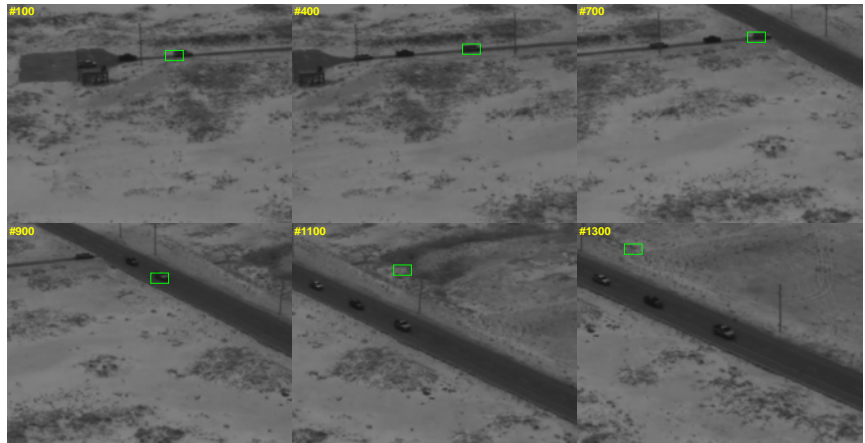
Figure 2. Tracking results using GMM tracker. There are no missing data in the video. Targets are lost in frames 1100 and 1300.
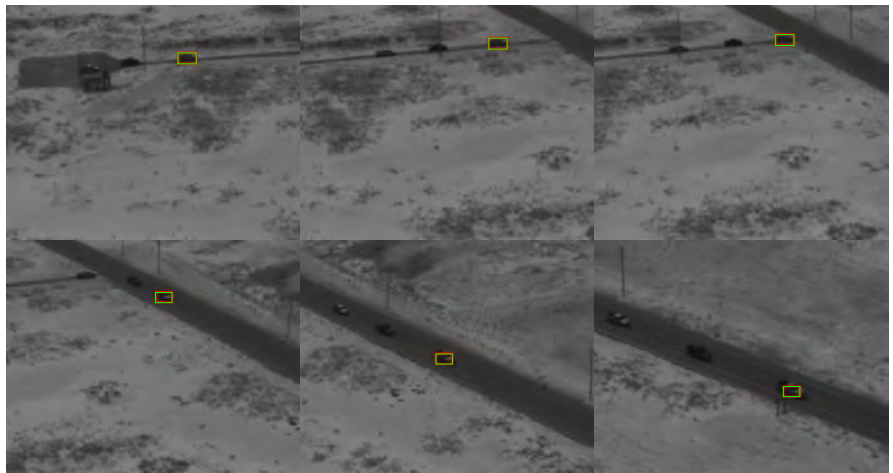
Figure 3. Tracking results using STAPLE. There is no missing data in the video.
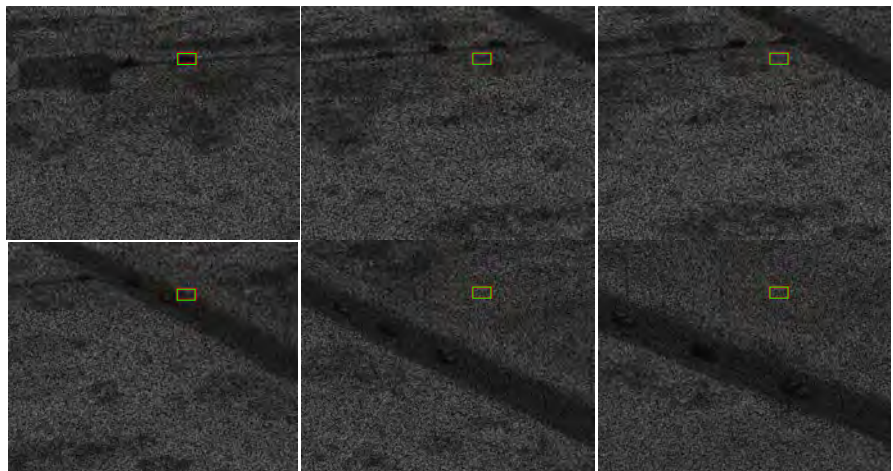
Figure 4. Tracking results using STAPLE. 50% of pixels are missing data in the video. Targets are lost in many frames.

and Video 5 for testing. Tables 1-3 show the performance metrics for different

172

missing pixel cases. Our first observation is that the number of frames with detection decreases when we have more missing pixels. This is reasonable. For those frames with detection, it can be seen that the CLE values increase when we have more missing pixels. This is also reasonable. The DP and EinGT values are all close to 100% if we have detection. **Figures 5-7** show the detection/tracking results in some selected frames. It can be seen that there are more missed detections in those cases of high missing rates.

Table 1. Tracking metrics for 0% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.96	1	1	2623/2678
Frontier	4.52	1	1	2422/2678
Silverado	4.81	1	1	2202/2678

Table 2. Tracking metrics for 50% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	5.75	1	1	2532/2678
Frontier	5.71	1	1	2371/2678
Silverado	5.24	1	1	1892/2678

Table 3. Tracking metrics for 75% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	6.3	1	1	1897/2678
Frontier	6.32	1	1	1933/2678
Silverado	5.28	1	1	831/2678

Figure 5. Tracking results for frames 1, 446, 892, 1338, 1784, and 2677. 0% missing case. Train using Video 4 and test using Video 5.
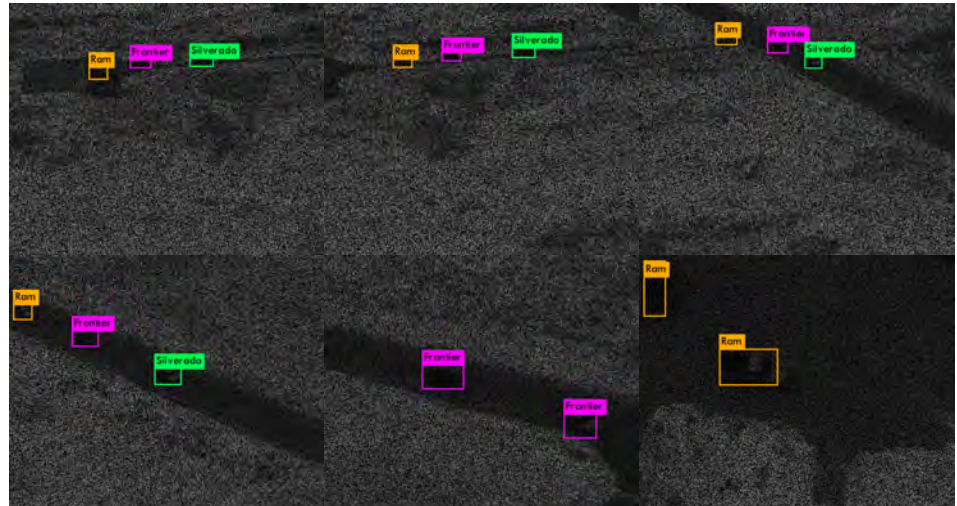
Figure 6. Tracking results for frames 1, 446, 892, 1338, 1784, and 2677. 50% missing case. Train using Video 4 and test using Video 5.
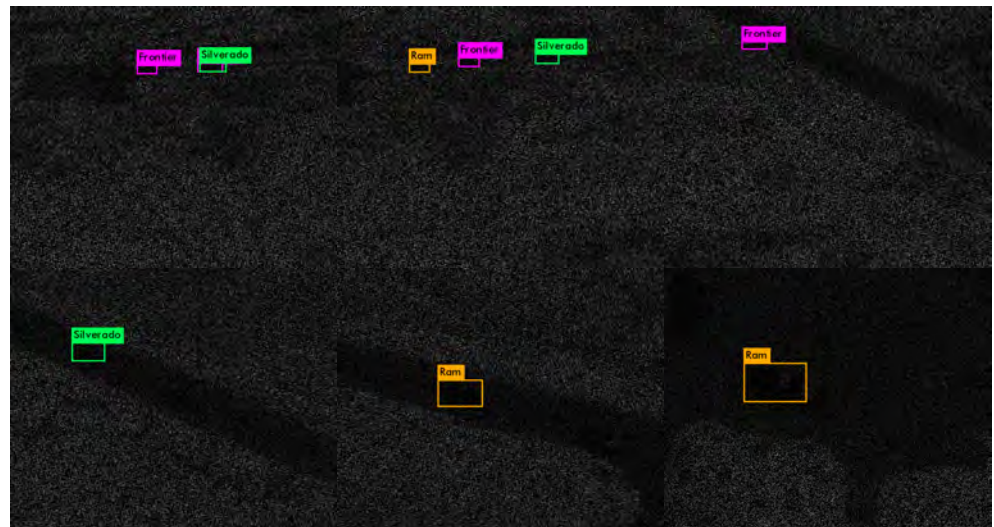
Figure 7. Tracking results for frames 1, 446, 892, 1338, 1784, and 2677. 75% missing case. Train using Video 4 and test using Video 5.

Tracking Results: Train using Video 5 and Test using Video 4

Tables 4-6 show the metrics when we used Video 5 for training and Video 4 for testing. We can see that the numbers of frames with detection are high for low missing rates. For frames with detection, the CLE values generally increase whereas the DP and EinGT values are relatively stable.

Figures 8-10 show the tracking results visually. It can be seen that we have some false detections in the parking lot area. However, when the targets are far away, the tracking appears to be good.

3.2. Classification Results

To illustrate the difficulty of classifying the three trucks, we include the pictures of them below in **Figure 11**. It can be seen that all of them have four doors and open

Table 4. Tracking metrics for 0% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.99	1	1	3282/3327
Frontier	4.09	1	0.99	3339/3327
Silverado	3.94	1	1	2012/3327

Table 5. Tracking metrics for 50% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	5.57	1	1	3247/3327
Frontier	4.2	1	0.99	3334/3327
Silverado	4.19	1	1	2002/3327

Table 6. Tracking metrics for 75% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	7	1	1	3075/3327
Frontier	4.62	1	0.99	3248/3327
Silverado	4.89	1	0.99	1864/3327

Figure 8. Tracking results for frames 1, 555, 1110, 1665, 2220, 3197. 0% missing case. Train using Video 5 and test using Video 4.

trunks. From a distance, it will be quite difficult to recognize them correctly.

For vehicle classification, we deployed two approaches: YOLO and ResNet. The YOLO comes with a default classifier. For the ResNet classifier, we performed customized training where the training data are augmented with rotation, scaling, and illumination variations.

Classification Results Using Video 4 for Training and Video 5 for testing

Classification is only applied to frames with detection of targets from the tracker. Tables 7-9 summarize the comparison between YOLO and ResNet classifiers for 0%, 50%, and 75% missing cases, respectively. We have two observations.

Figure 9. Tracking results for frames 1, 555, 1110, 1665, 2220, 3197. 50% missing case. Train using Video 5 and test using Video 4.
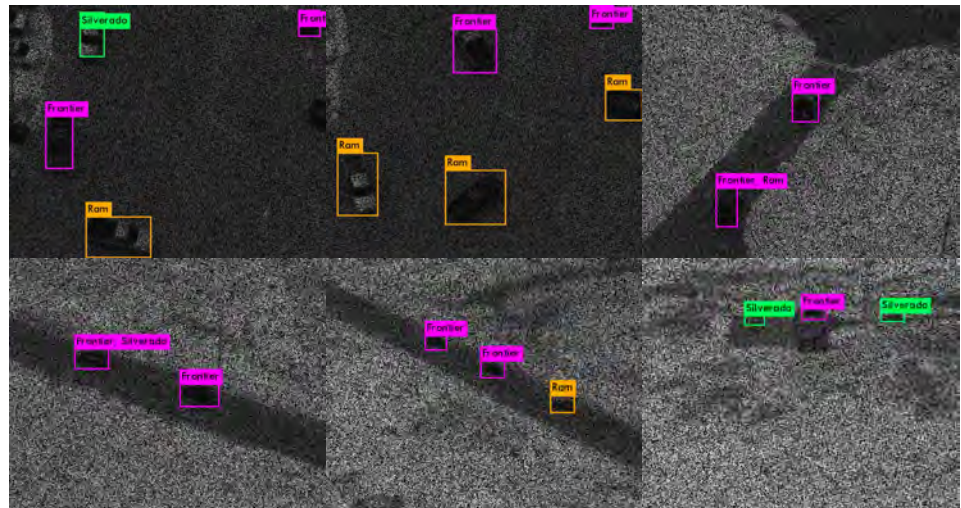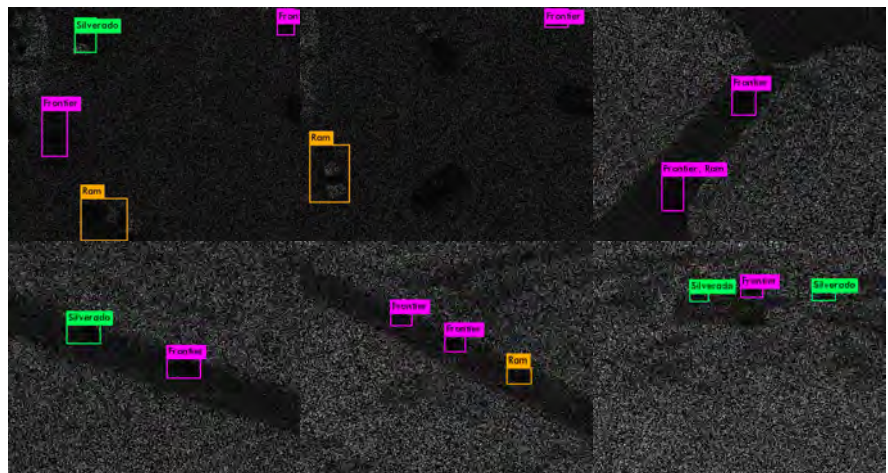

Figure 10. Tracking results for frames 1, 555, 1110, 1665, 2220, 3197. 75% missing case. Train using Video 5 and test using Video 4.

Figure 11. Pictures of Ram, Frontier, and Silverado.

First, the YOLO classifier outputs are worse than those of the ResNet. Second, when missing rates increase, the classification accuracy drops.

Classification Results Using Video 5 for training and Video 4 for testing

As shown in Tables 10-12, the ResNet classifier has much better performance than that of YOLO. Moreover, the classification results using ResNet are still quite good for 75% missing case.

Table 7. Classification results for 0% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	381	265	1953	0.1466
	Frontier	202	2196	0	0.9158
	Silverado	2132	0	65	0.0296

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	2220	32	371	0.8464
	Frontier	161	2223	38	0.9178
	Silverado	987	41	1174	0.5332

Table 8. Classification results for 50% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier output. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	299	247	1986	0.1181
	Frontier	334	1998	16	0.8509
	Silverado	1823	8	51	0.0271

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	1703	56	773	0.6726
	Frontier	158	2021	192	0.8524
	Silverado	543	43	1306	0.6903

3.3. Discussions

We are interested in the tracking and classification performance in the 75% missing data case because only 25% of pixels need to be stored and transmitted. At this missing rate, using the numbers shown in Table 13, the averaged percentages of frames being detected are 58% for testing using Video 5 and 82% for testing using Video 4, respectively. From Table 14, the averaged percentages of classification are 60% for testing using Video 5 and 78% for testing using Video 4, respectively.

Table 9. Classification results for 75% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	183	234	1479	0.0965
Predicted	Frontier	459	1360	106	0.7065
	Silverado	710	91	28	0.0338

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1032	590	275	0.5440
Predicted	Frontier	132	1722	79	0.8908
	Silverado	331	190	310	0.3730

Table 10. Classification results for 0% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	601	1480	1157	0.1856
Predicted	Frontier	83	3151	0	0.9743
	Silverado	1496	44	435	0.2203

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	2837	72	373	0.8644
Predicted	Frontier	670	2514	155	0.7529
	Silverado	415	18	1579	0.7848

4. Tracking and Classification Results Using MWIR Videos

Similar to the SWIR videos, we have also two MWIR videos from our sponsor. In Section 4.1, we present the conventional and our proposed tracking results. Section 4.2 shows the classification results.

4.1. Tracking Results

Conventional Tracking Results

Here, we only include the STAPLE results because GMM tracker did not work

Table 11. Classification results for 50% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	596	1376	1221	0.1867
Predicted	Frontier	191	2998	0	0.9401
	Silverado	1484	30	464	0.2346

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	2062	227	958	0.6350
Predicted	Frontier	212	2989	133	0.8965
	Silverado	26	10	1966	0.9820

Table 12. Classification results for 75% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	506	1222	1316	0.1662
Predicted	Frontier	334	2804	0	0.8936
	Silverado	1352	13	490	0.2642

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	2300	120	655	0.7480
Predicted	Frontier	740	2392	116	0.7365
	Silverado	250	6	1608	0.8627

at all. STAPLE appears to work reasonably well for zero and 50% missing rate cases (Figure 12 and Figure 13). When the missing rate increases to 75%, the STAPLE tracker failed completely as shown in Figure 14. It is observed that one issue with STAPLE is that it is difficult for it to track multiple vehicles simultaneously.

MWIR Results: Train using Video 4 and Test using Video 5

Here, we used Video 4 for training and Video 5 for testing. Tables 15-17

Table 13. Tracking metrics for 75% missing case. (a) Train using Video 4 and test using Video 5; (b) Train using Video 5 and test using Video 4.

(a)

	Number of frames with detection
Ram	1897/2678
Frontier	1933/2678
Silverado	831/2678

(b)

	Number of frames with detection
Ram	3075/3327
Frontier	3248/3327
Silverado	1864/3327

Table 14. ResNet classification at 75% missing rate. (a) Train using Video 4 and test using Video 5; (b) Train using Video 5 and test using Video 4.

(a)

	Classification accuracy
Ram	0.5440
Frontier	0.8908
Silverado	0.3730

(b)

	Classification accuracy
Ram	0.7480
Frontier	0.7365
Silverado	0.8627

Figure 12. Tracking results for frames STAPLE at 0% missing data; 1, 555, 1109, 1663, 2217, 2771.

show the performance metrics. Our first observation is that the number of frames with detection decreases when we have more missing pixels. This is reasonable.
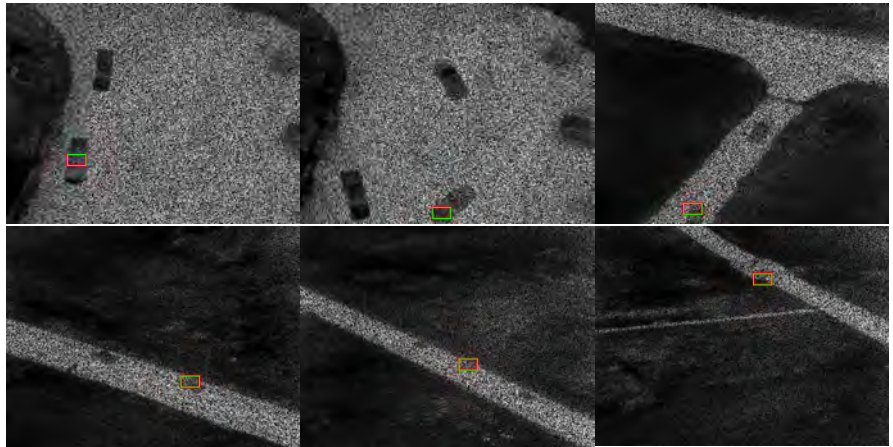
Figure 13. Tracking results for frames STAPLE at 50% missing; 1, 555, 1109, 1663, 2217, 2771.

Figure 14. Tracking results for frames STAPLE at 75% missing; 1, 555, 1109, 1663, 2217, 2771.
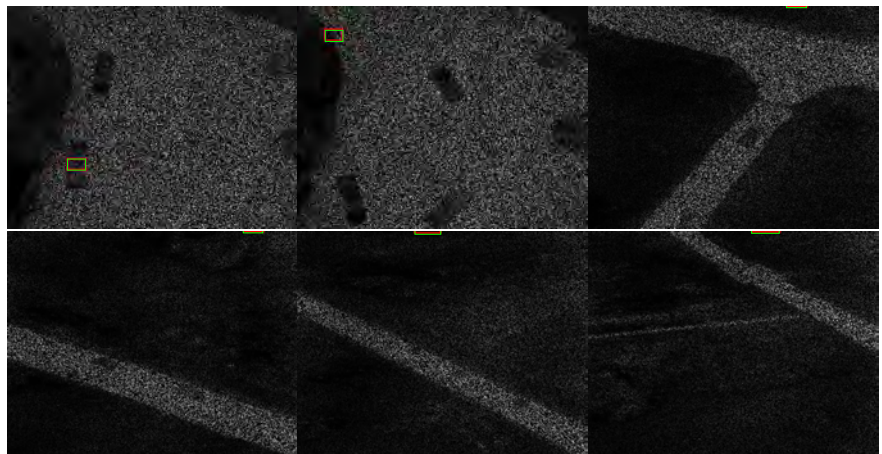
Table 15. MWIR tracking metrics for 0% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	3.14	1	1	2568/2677
Frontier	3.02	1	0.99	2671/2677
Silverado	4.82	1	0.84	2461/2677

Table 16. MWIR tracking metrics for 50% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.88	1	1	2465/2677
Frontier	4.69	1	0.99	2650/2677
Silverado	4.7	1	0.93	2124/2677

For those frames with detection, it can be seen that the CLE values increase when we have more missing pixels. This is also reasonable. The DP and EinGT values are all close to 100% if we have detection. **Figures 15-18** show the tracking results in some selected frames. It can be seen that there are more missed detections in those cases of high missing rates. The labels come from the YOLO tracker outputs and have more errors when the missing rates are high.
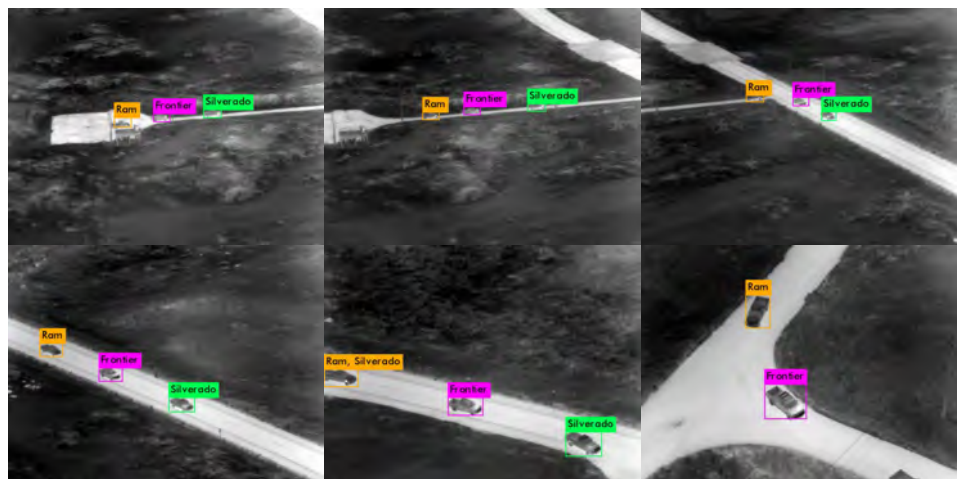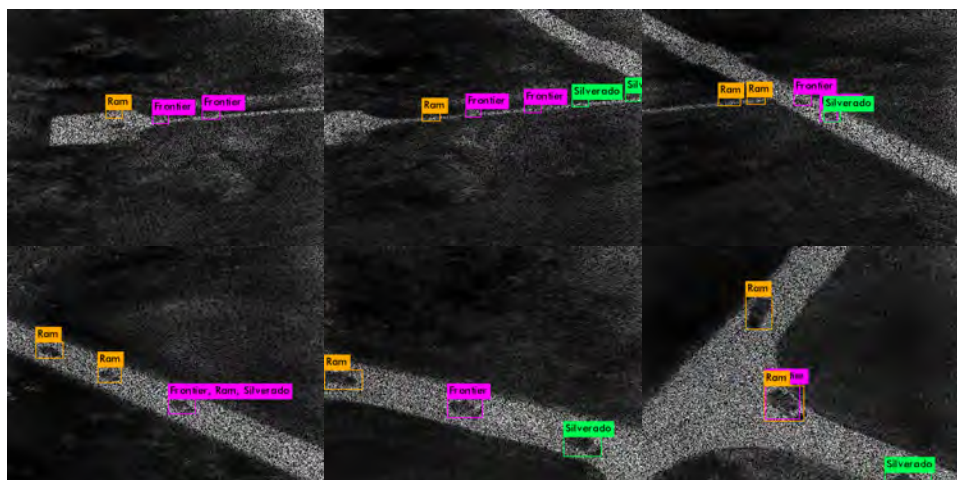
Table 17. MWIR tracking metrics for 75% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	6.29	1	0.99	1917/2677
Frontier	5.8	1	0.99	1705/2677
Silverado	7.13	1	0.93	1453/2677

Figure 15. MWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 0% missing case. Train using Video 4 and test using Video 5.

Figure 16. MWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 50% missing case. Train using Video 4 and test using Video 5.

MWIR Results: Train using Video 5 and Test using Video 4

Tables 18-20 show the metrics when we used Video 5 for training and Video 4 for testing. We can see that the numbers of frames with detection are high for low missing rates. For frames with detection, the CLE values generally increase whereas the DP and EinGT values are relatively stable. Figures 18-20 show the tracking results visually. It can be seen that we have some false detections in the parking lot area. However, when the targets are far away, the tracking appears to
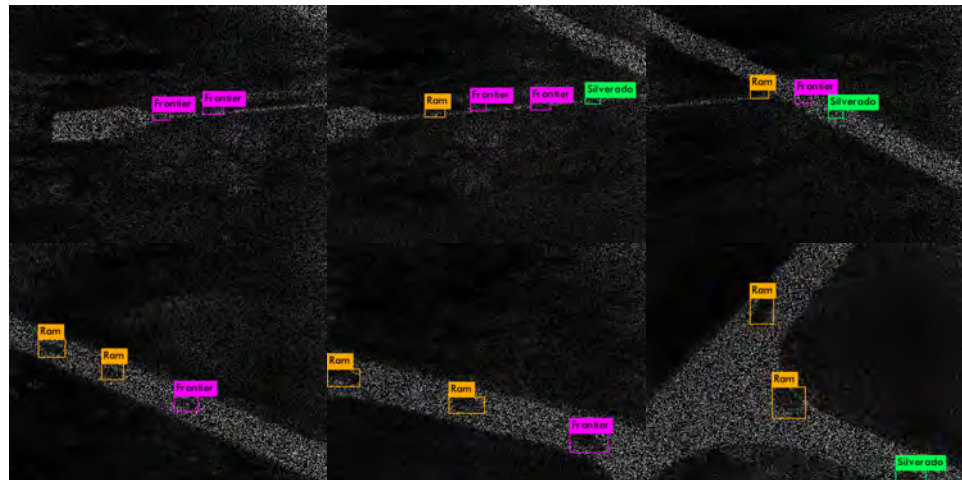

Figure 17. MWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 75% missing case. Train using Video 4 and test using Video 5.

Figure 18. MWIR tracking results for frames 1, 555, 1109, 1663, 2217, and 2771. 0% missing case. Train using Video 5 and test using Video 4.

Table 18. MWIR tracking metrics for 0% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.18	1	0.96	2858/3324
Frontier	4.05	1	0.98	3234/3324
Silverado	5.2	1	0.99	2027/3324

Table 19. MWIR tracking metrics for 50% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.8	1	0.96	2755/3324
Frontier	5.36	1	0.97	3134/3324
Silverado	5.56	1	0.98	1860/3324

Table 20. MWIR tracking metrics for 75% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	6.09	1	0.94	2295/3324
Frontier	6.63	1	0.94	2108/3324
Silverado	6.28	1	0.96	1615/3324
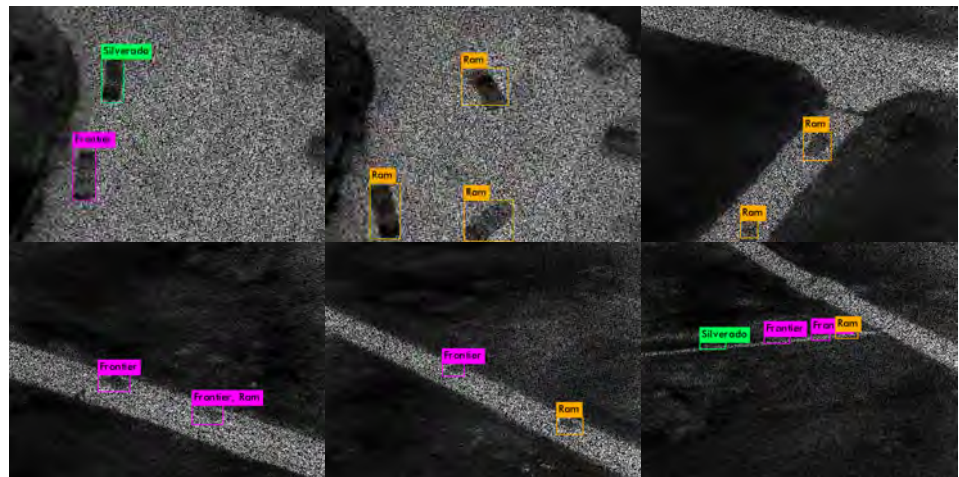
Figure 19. MWIR tracking results for frames 1, 555, 1109, 1663, 2217, and 2771. 50% missing case. Train using Video 5 and test using Video 4.
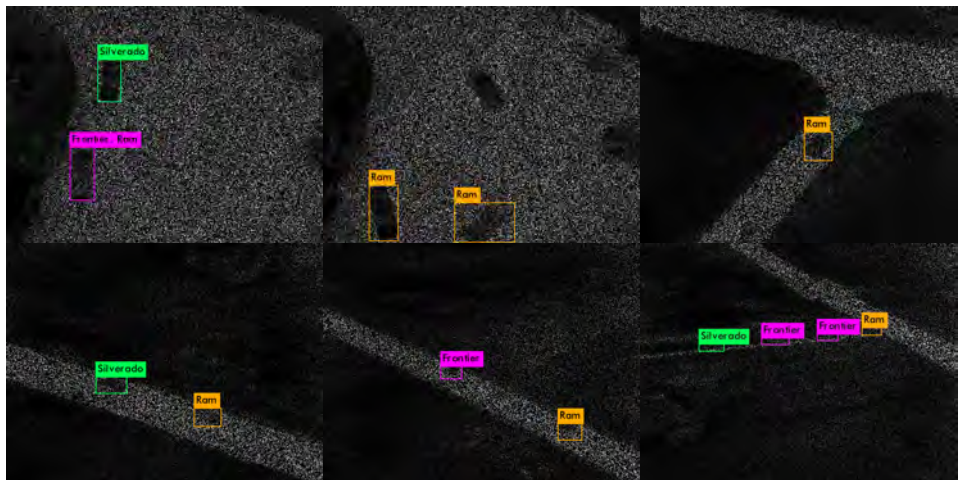
Figure 20. MWIR tracking results for frames 1, 555, 1109, 1663, 2217, and 2771. 75% missing case. Train using Video 5 and test using Video 4.

be good. The labels come from the YOLO tracker. We will see in the next section that the ResNet classifier has better performance than that of YOLO.

4.2. Classification Results

MWIR Classification Results Using Video 4 for Training and Video 5 for testing

Classification is only applied to frames with detection of targets from the tracker. Tables 21-23 summarize the comparison between YOLO and ResNet classifiers for 0%, 50%, and 75% missing cases, respectively. We have two observations. First, the YOLO classifier outputs are worse than those of the ResNet.

Table 21. Classification results for 0% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	188	732	1648	0.0732
Predicted	Frontier	25	2404	237	0.9017
	Silverado	2201	7	160	0.0676

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	2367	87	114	0.9217
Predicted	Frontier	275	2371	25	0.8877
	Silverado	1444	274	743	0.3019

Table 22. Classification results for 50% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier output. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	424	877	872	0.1951
Predicted	Frontier	705	1404	91	0.6382
	Silverado	1660	70	39	0.0220

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1630	44	791	0.6613
Predicted	Frontier	829	1595	226	0.6019
	Silverado	1245	82	797	0.3752

Second, when missing rates increase, the classification accuracy drops.

MWIR Classification Results Using Video 5 for training and Video 4 for testing

As shown in Tables 24-26, the ResNet classifier has much better performance than that of YOLO. Moreover, the classification results using ResNet are still quite good for 75% missing case.

Table 23. Classification results for 75% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	276	1078	549	0.1450
Predicted	Frontier	559	1121	8	0.6641
	Silverado	1361	66	0	0.0000

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1267	269	381	0.6609
Predicted	Frontier	572	1103	30	0.6469
	Silverado	945	224	284	0.1955

Table 24. Classification results for 0% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	528	735	1544	0.1884
Predicted	Frontier	1006	2093	0	0.6754
	Silverado	1429	61	532	0.2631

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1934	65	859	0.6767
Predicted	Frontier	114	3041	79	0.9403
	Silverado	535	5	1487	0.7336

Table 25. Classification results for 50% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	444	933	1329	0.1641
Predicted	Frontier	958	1993	103	0.6526
	Silverado	1319	21	518	0.2788

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1993	85	677	0.7234
Predicted	Frontier	112	2898	124	0.9247
	Silverado	839	20	1001	0.5382

Table 26. Classification results for 75% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results; (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	270	691	1303	0.1193
Predicted	Frontier	622	1318	144	0.6324
	Silverado	1030	43	540	0.3348

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1325	205	765	0.5773
Predicted	Frontier	154	1867	87	0.8857
	Silverado	652	98	865	0.5356

4.3. Discussions

Similar to the SWIR study, we are interested in the tracking and classification performance in the 75% missing data case where one can have fewer pixels to save and transmit. At this missing rate, using the numbers shown in Table 27, the averaged percentages of frames being detected are 63% for testing using Video 5 and 60% for testing using Video 4, respectively. From Table 28, the

Table 27. Tracking metrics for 75% missing case. (a) Train using Video 4 and test using Video 5. (b) Train using Video 5 and test using Video 4.

(a)

	Number of frames with detection
Ram	1917/2677
Frontier	1705/2677
Silverado	1453/2677

(b)

	Number of frames with detection
Ram	2295/3324
Frontier	2108/3324
Silverado	1615/3324

Table 28. ResNet classification at 75% missing rate. (a) Train using Video 4 and test using Video 5. (b) Train using Video 5 and test using Video 4.

(a)

	Classification accuracy
Ram	0.6609
Frontier	0.6469
Silverado	0.1955

(b)

	Classification accuracy
Ram	0.5773
Frontier	0.8857
Silverado	0.5356

averaged percentages of classification are 50% for testing using Video 5 and 66% for testing using Video 4, respectively.

5. Tracking and Classification Results Using LWIR Videos

In this section, we summarize the tracking and classification results using LWIR videos.

### 5.1. Tracking Results

Conventional Tracker Results

We first present tracking results using STAPLE. Similar to the SWIR and MWIR cases, STAPLE did not perform well for the various cases as shown in Figures 21-23.

LWIR Results: Train using Video 4 and Test using Video 5

Tables 29-31 show the tracking results for different missing cases. The missed detection rates increase as more pixels are missing. From Figures 24-26, the

Figure 21. STAPLE LWIR 0% missing.
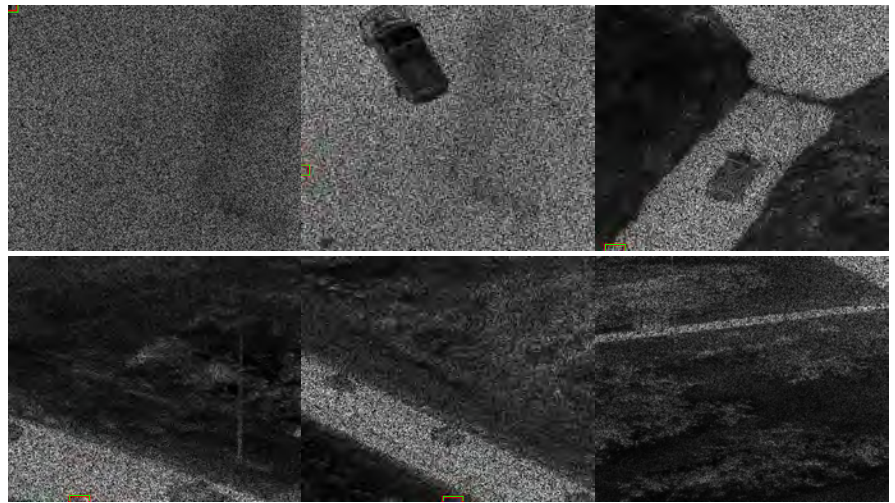
Figure 22. STAPLE LWIR 50% missing.

Table 29. LWIR tracking metrics for 0% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	4.08	1	1	1606/2677
Frontier	4.51	1	0.98	2136/2677
Silverado	5.14	1	0.99	735/2677

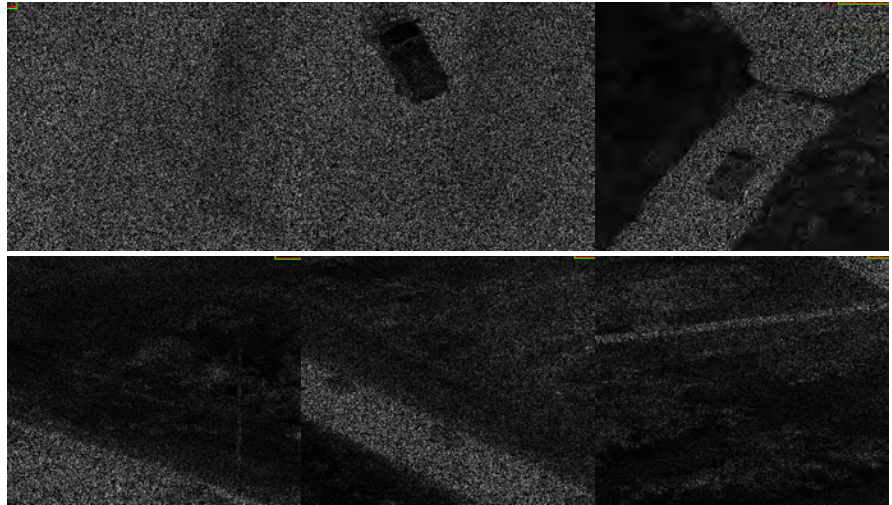**Table 30.** LWIR tracking metrics for 50% missing case. Train using Video 4 and test using Video 5.

|	CLE	DP	EinGT	Number of frames with detection
Ram	4.65	1	1	1602/2677
Frontier	5.04	1	0.98	2084/2677
Silverado	5.38	1	0.99	648/2677

tracking results are quite good except that the labels from the YOLO tracker have some wrong labels.

LWIR Results: Train using Video 5 and Test using Video 4

From Tables 32-34 and Figures 27-29, we have the same observations here as the earlier sections. That is, as missing rates increase, the tracking performance drops.
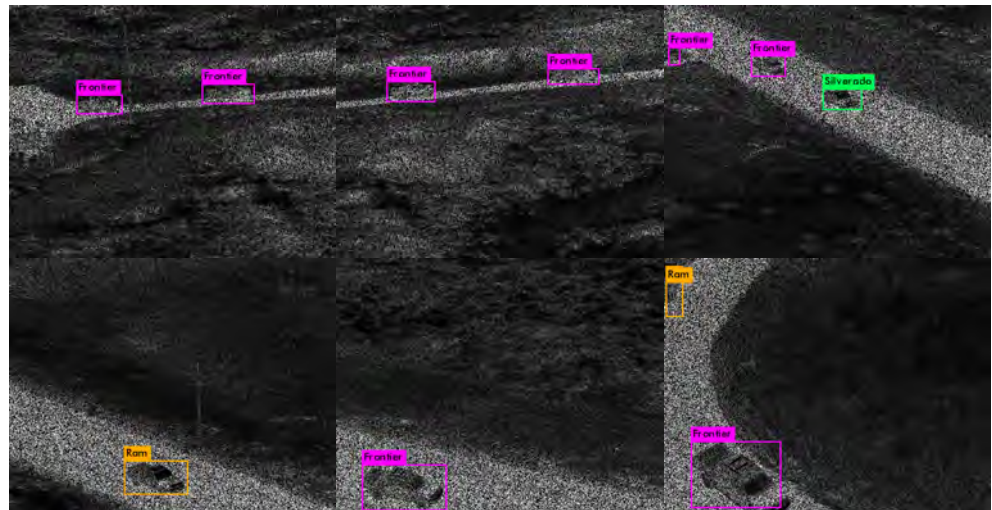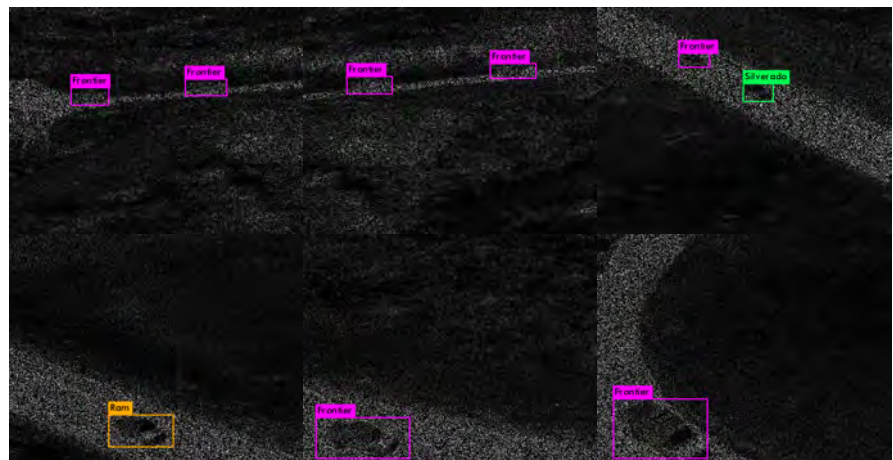

Figure 23. STAPLE LWIR 75% missing.
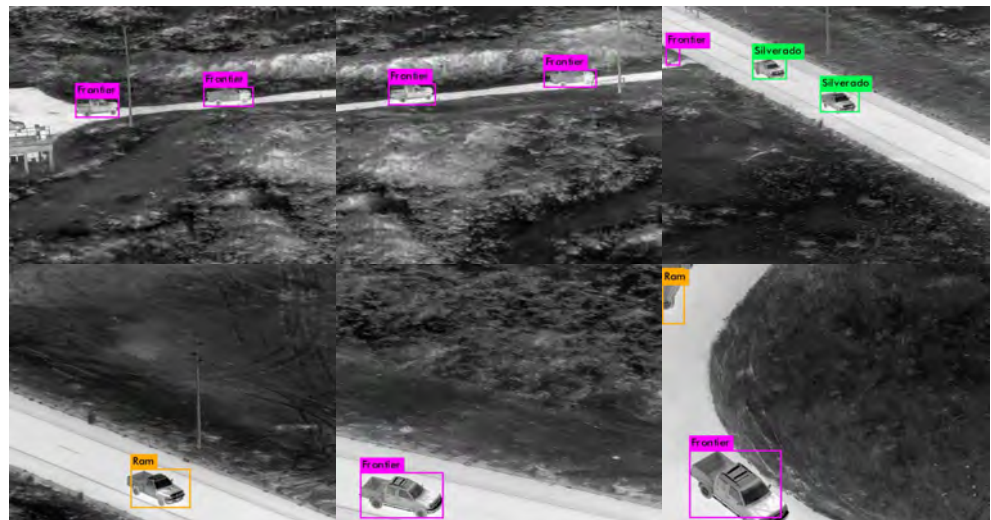
Figure 24. LWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 0% missing case. Train using Video 4 and test using Video 5.

Table 31. LWIR tracking metrics for 75% missing case. Train using Video 4 and test using Video 5.

	CLE	DP	EinGT	Number of frames with detection
Ram	5.48	1	1	1427/2677
Frontier	6.47	1	0.98	1235/2677
Silverado	4,85	1	0.99	489/2677

Figure 25. LWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 50% missing case. Train using Video 4 and test using Video 5.
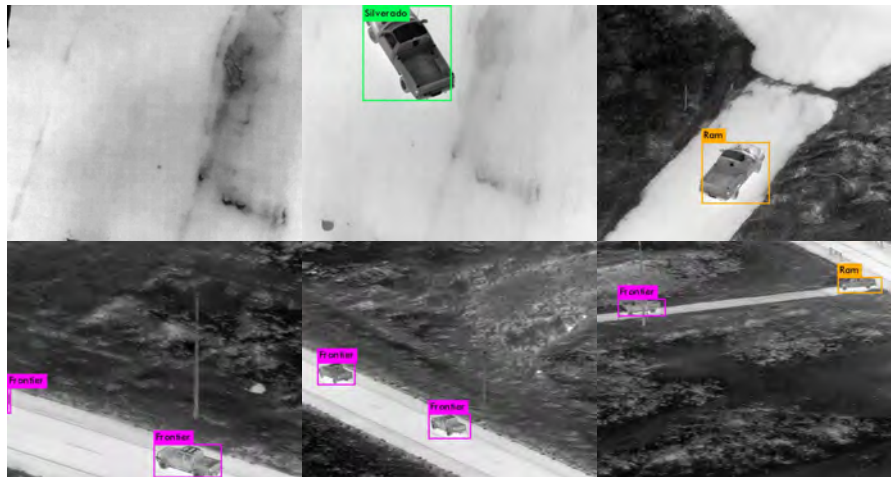

Figure 26. LWIR tracking results for frames 1, 447, 893, 1339, 1785, and 2231. 75% missing case. Train using Video 4 and test using Video 5.

Table 32. LWIR tracking metrics for 0% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	6.37	1	0.96	1635/3303
Frontier	4.22	1	0.98	1902/3303
Silverado	4.17	1	1	745/3303

Table 33. LWIR tracking metrics for 50% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	7.56	1	0.96	1373/3303
Frontier	5.52	1	0.98	1774/3303
Silverado	6.62	1	1	599/3303

Figure 27. LWIR tracking results for frames 1, 551, 1101, 1651, 2201, and 2751. 0% missing case. Train using Video 5 and test using Video 4.

Table 34. LWIR tracking metrics for 75% missing case. Train using Video 5 and test using Video 4.

	CLE	DP	EinGT	Number of frames with detection
Ram	9.75	1	0.96	557/3303
Frontier	6.88	1	0.96	805/3303
Silverado	7.54	1	1	238/3303
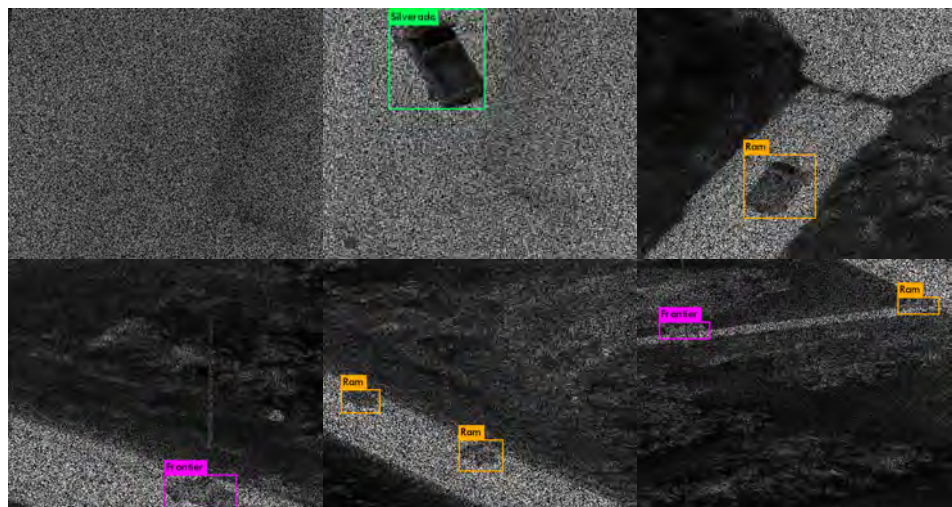
Figure 28. LWIR tracking results for frames 1, 551, 1101, 1651, 2201, and 2751. 50% missing case. Train using Video 5 and test using Video 4.

5.2. Classification Results

LWIR Classification Results Using Video 4 for Training and Video 5 for testing

Here, from Tables 35-37, we observe that ResNet results are better than YOLO. Even for high missing rates, the ResNet performs reasonably well.
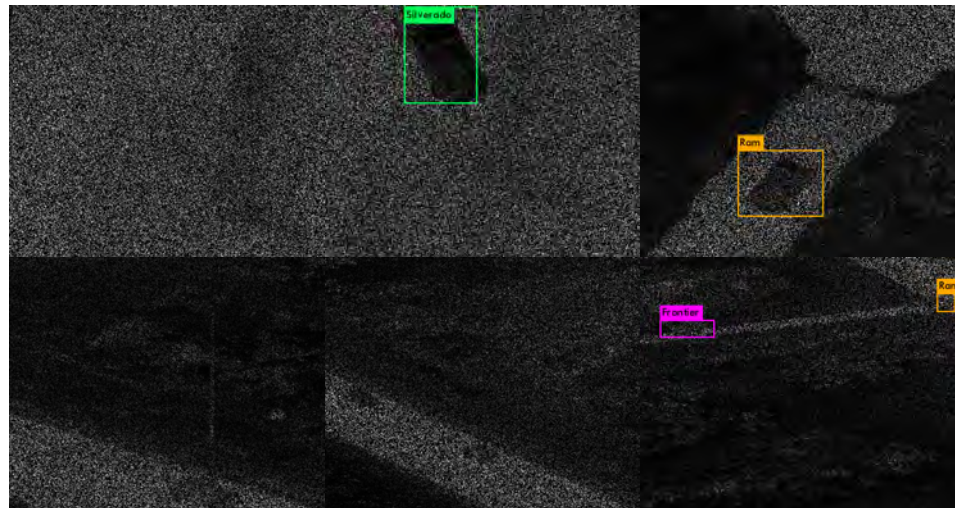
Figure 29. LWIR tracking results for frames 1, 551, 1101, 1651, 2201, and 2751. 75% missing case. Train using Video 5 and test using Video 4.

Table 35. Classification results for 0% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	302	841	370	0.1996
Predicted	Frontier	362	1634	49	0.7990
	Silverado	230	459	24	0.0337

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1606	0	0	1.0000
Predicted	Frontier	766	1369	1	0.6409
	Silverado	285	32	418	0.5687

LWIR Classification Results Using Video 5 for training and Video 4 for testing

From Tables 38-40 below, we have similar observations as the earlier section. ResNet performs quite well for LWIR case.

5.3. Discussions

Similar to the SWIR study, we are interested in the tracking and classification performance in the 75% missing data case where one can have fewer pixels to save and transmit. At this missing rate, using the numbers shown in Table 41, the averaged percentages of frames being detected are 43% for testing using Video 5 and 16% for testing using Video 4, respectively. The detection percentages appear

Table 36. Classification results for 50% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier output. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	442	687	425	0.2844
Predicted	Frontier	417	1525	59	0.7621
	Silverado	146	437	48	0.0761

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1598	3	1	0.9975
Predicted	Frontier	789	1291	4	0.6195
	Silverado	249	29	370	0.5710

Table 37. Classification results for 75% missing case. Video 4 for training and Video 5 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	446	670	292	0.3168
Predicted	Frontier	425	1273	20	0.7410
	Silverado	35	443	8	0.0165

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
	Ram	1414	13	0	0.9909
Predicted	Frontier	60	1174	1	0.9506
	Silverado	179	65	245	0.5010

to be low. This is mainly because, for LWIR videos, each frame contains roughly one to two vehicle per frame whereas in the SWIR and MWIR videos, we have multiple vehicles in each frame. From Table 42, the averaged percentages of classification are 81% for testing using Video 5 and 79% for testing using Video 4, respectively.

6. Conclusions

We present a deep learning approach for multiple target tracking and classification using infrared videos (SWIR, MWIR, and LWIR) directly in the compressive measurement domain. Key advantages include fast processing without time

Table 38. Classification results for 0% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	550	620	329	0.3669
	Frontier	104	1773	0	0.9446
	Silverado	506	106	62	0.0920

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	1605	24	6	0.9817
	Frontier	37	1790	75	0.9411
	Silverado	138	29	578	0.7758

Table 39. Classification results for 50% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	599	472	211	0.4672
	Frontier	258	1479	2	0.8505
	Silverado	531	9	53	0.0894

(b)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	957	249	167	0.6970
	Frontier	24	1742	8	0.9820
	Silverado	217	111	271	0.4524

Table 40. Classification results for 75% missing case. Video 5 for training and Video 4 for testing. (a) YOLO classifier outputs. Left is the confusion matrix; right is the classification results. (b) ResNet classifier outputs. Left is the confusion matrix; right is the classification results.

(a)

		Actual			Classification Accuracy
		Ram	Frontier	Silverado	
Predicted	Ram	223	203	118	0.4099
	Frontier	58	747	0	0.9280
	Silverado	192	2	40	0.1709

(b)

		Actual		
		Ram	Frontier	Silverado	Classification Accuracy
Predicted	Ram	360	134	63	0.6463
	Frontier	9	794	2	0.9863
	Silverado	48	15	175	0.7353

Table 41. Tracking metrics for 75% missing case. (a) Train using Video 4 and test using Video 5; (b) Train using Video 5 and test using Video 4.

(a)

	Number of frames with detection
Ram	1635/3303
Frontier	1902/3303
Silverado	745/3303

(b)

	Number of frames with detection
Ram	557/3303
Frontier	805/3303
Silverado	238/3303

Table 42. ResNet classification at 75% missing rate. (a) Train using Video 4 and test using Video 5; (b) Train using Video 5 and test using Video 4.

(a)

	Classification accuracy
Ram	0.9909
Frontier	0.9506
Silverado	0.5010

(b)

	Classification accuracy
Ram	0.6463
Frontier	0.9863
Silverado	0.7353

consuming image reconstruction. Experiments using various types of infrared videos clearly demonstrated the performance of the proposed approach under different conditions even when the training data are limited. Moreover, comparison with conventional trackers showed that the deep learning based approach is much more accurate, especially when the missing rate is high.

One future direction is to integrate the proposed approach with video cameras and perform real-time tracking and classification.

Acknowledgements

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Li, X., Kwan, C., Mei, G. and Li, B. (2006) A Generic Approach to Object Matching and Tracking. In: Campilho, A. and Kamel, M.S., Eds., *Image Analysis and Recognition. ICIAR* 2006. *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 839-849. https://doi.org/10.1007/11867586_76

[2] Zhou, J. and Kwan, C. (2018) Tracking of Multiple Pixel Targets Using Multiple Cameras. In: Huang, T., Lv, J., Sun, C. and Tuzikov, A., Eds., *Advances in Neural Networks. Lecture Notes in Computer Science*, Springer, Cham, 484-493. https://doi.org/10.1007/978-3-319-92537-0_56

[3] Zhou, J. and Kwan, C. (2018) Anomaly Detection in Low Quality Traffic Monitoring Videos Using Optical Flow. *Proceedings of SPIE* 10649, *Pattern Recognition and Tracking XXIX*, 106490F.

[4] Kwan, C., Zhou, J., Wang, Z. and Li, B. (2018) Efficient Anomaly Detection Algorithms for Summarizing Low Quality Videos. *Proceedings of SPIE* 10649, *Pattern Recognition and Tracking XXIX*, 1064906. https://doi.org/10.1117/12.2303764

[5] Kwan, C., Chou, B. and Kwan, L. M. (2018) A Comparative Study of Conventional and Deep Learning Target Tracking Algorithms for Low Quality Videos. In: Huang, T., Lv, J., Sun, C. and Tuzikov, A., Eds., *Advances in Neural Networks. Lecture Notes in Computer Science*, Springer, Cham, 521-531. https://doi.org/10.1007/978-3-319-92537-0_60

[6] Kwan, C., Yin, J. and Zhou, J. (2018) The Development of a Video Browsing and Video Summary Review Tool. *Proceedings of SPIE* 10649, *Pattern Recognition and Tracking XXIX*, 1064907. https://doi.org/10.1117/12.2303654

[7] Zhao, Z., Chen, H., Chen, G., Kwan, C. and Li, X.R. (2006) IMM-LMMSE Filtering Algorithm for Ballistic Target Tracking with Unknown Ballistic Coefficient. *Proceedings of SPIE*, *Volume* 6236, *Signal and Data Processing of Small Targets*. https://doi.org/10.1117/12.665760

[8] Zhao, Z., Chen, H., Chen, G., Kwan, C. and Li, X.R. (2006) Comparison of Several Ballistic Target Tracking Filters. *Proceedings of American Control Conference*, Minneapolis, MN, 14-16 June 2006, 2197-2202.

[9] Candes, E.J. and Wakin, M.B. (2008) An Introduction to Compressive Sampling. *IEEE Signal Processing Magazine*, **25**, 21-30. https://doi.org/10.1109/MSP.2007.914731

[10] Kwan, C., Chou, B., Echavarren, A., Budavari, B., Li, J. and Tran, T. (2018) Compressive Vehicle Tracking Using Deep Learning. *IEEE Ubiquitous Computing, Electronics & Mobile Communication Conference*, New York City, 8-10 November 2018, 51-56. https://doi.org/10.1109/UEMCON.2018.8796778

[11] Tropp, J.A. (2004) Greed Is Good: Algorithmic Results for Sparse Approximation. *IEEE Transactions on Information Theory*, **50**, 2231-2242. https://doi.org/10.1109/TIT.2004.834793

[12] Yang, J. and Zhang, Y. (2011) Alternating Direction Algorithms for L1-Problems in Compressive Sensing. *SIAM Journal on Scientific Computing*, **33**, 250-278. https://doi.org/10.1137/090777761

[13] Dao, M., Kwan, C., Koperski, K. and Marchisio, G. (2017) A Joint Sparsity Approach to Tunnel Activity Monitoring Using High Resolution Satellite Images. *2017 IEEE* 8*th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference*, New York, 19-21 October 2017, 322-328. https://doi.org/10.1109/UEMCON.2017.8249061

[14] Zhou, J., Ayhan, B., Kwan, C. and Tran, T. (2018) ATR Performance Improvement Using Images with Corrupted or Missing Pixels. *Proceedings of SPIE* 10649, *Pattern Recognition and Tracking XXIX*, 106490E.

[15] Applied Research LLC (2017) Phase 1 Final Report.

[16] Kwan, C., Chou, B., Yang, J. and Tran, T. (2019) Target Tracking and Classification Directly in Compressive Measurement for Low Quality Videos. *Pattern Recognition and Tracking XXX* (*Conference SI*120). https://doi.org/10.1117/12.2518496

[17] Kwan, C., Gribben, D. and Tran, T. (2019) Multiple Human Objects Tracking and Classification Directly in Compressive Measurement Domain for Long Range Infrared Videos. *IEEE Ubiquitous Computing, Electronics & Mobile Communication Conference*, New York City.

[18] Kwan, C., Chou, B., Yang, J., and T. Tran, T. (2019) Deep Learning Based Target Tracking and Classification Directly in Compressive Measurement for Low Quality Videos. *Signal & Image Processing: An International Journal* (*SIPIJ*).

[19] Kwan, C., Gribben, D. and Tran, T. (2019) Tracking and Classification of Multiple Human Objects Directly in Compressive Measurement Domain for Low Quality Optical Videos. *IEEE Ubiquitous Computing, Electronics & Mobile Communication Conference*, New York City.

[20] Kwan, C., Chou, B., Yang, J., Rangamani, A., Tran, T., Zhang, J. and Etienne-Cummings, R. (2019) Target Tracking and Classification Directly Using Compressive Sensing Camera for SWIR Videos. *Signal, Image, and Video Processing*, **13**, 1629-1637. https://doi.org/10.1007/s11760-019-01506-4

[21] Kwan, C., Chou, B., Yang, J., Rangamani, A., Tran, T., Zhang, J. and Etienne-Cummings, R. (2019) Target Tracking and Classification Using Compressive Measurements of MWIR and LWIR Coded Aperture Cameras. *Journal Signal and Information Processing*, **10**, 73-95. https://doi.org/10.4236/jsip.2019.103006

[22] Kwan, C., Chou, B., Yang, J., Rangamani, A., Tran, T., Zhang, J. and Etienne-Cummings, R., (2019) Deep Learning based Target Tracking and Classification for Low Quality Videos Using Coded Aperture Camera. *Sensors*, **19**, 3702. https://doi.org/10.3390/s19173702

[23] Yang, M.H., Zhang, K. and Zhang, L. (2012) Real-Time Compressive Tracking. In *European Conference on Computer Vision*.

[24] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep Residual Learning for Image Recognition. 2016 *Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 27-30 June 2016, 770-778. https://doi.org/10.1109/CVPR.2016.90

[25] Redmon, J. and Farhadi, A. (2018) YOLOv3: An Incremental Improvement.

[26] Ren S., He, K., Girshick, R. and Sun, J. (2015) Faster R-CNN: Towards Real-Time

Object Detection with Region Proposal Networks. In: *Advances in Neural Information Processing Systems*, 1-9.

[27] Kwan, C., Chou, B., Yang, J., Budavari, B., and Tran, T. (2019) Compressive Object Tracking and Classification Using Deep Learning for Infrared Videos. *Pattern Recognition and Tracking XXX* (*Conference SI* 120).
https://doi.org/10.1117/12.2518490

[28] Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O. and Torr, P. (2016) Staple: Complementary Learners for Real-Time Tracking. 2016 *Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 27-30 June 2016, 1401-1409.
https://doi.org/10.1109/CVPR.2016.156

[29] Stauffer, C. and Grimson, W.E.L. (1999) Adaptive Background Mixture Models for Real-Time Tracking, Computer Vision and Pattern Recognition. *IEEE Computer Society Conference*, **2**, 2246-2252.