

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

Shuo Liu, Zheng Liu

School of Engineering, Faculty of Applied Science, University of British Columbia (Okanagan), 1137  
Alumni Avenue Kelowna, BC V1V 1V7, Canada  
CANADA

zheng.liu@ubc.ca

### ABSTRACT

*Object Detection is critical for automatic military operations. However, the performance of current object detection algorithms is deficient in terms of the requirements in military scenarios. This is mainly because the object presence is hard to detect due to the indistinguishable appearance and dramatic changes of object's size which is determined by the distance to the detection sensors. Recent advances in deep learning have achieved promising results in many challenging tasks. The state-of-the-art in object detection is represented by convolutional neural networks (CNNs), such as the fast R-CNN algorithm. These CNN-based methods improve the detection performance significantly on several public generic object detection datasets. However, their performance on detecting small objects or undistinguishable objects in visible spectrum images is still insufficient. In this study, we propose a novel detection algorithm for military objects by fusing multi-channel CNNs. We combine spatial, temporal and thermal information by generating a three-channel image, and they will be fused as CNN feature maps in an unsupervised manner. The backbone of our object detection framework is from the fast R-CNN algorithm, and we utilize cross-domain transfer learning technique to fine-tune the CNN model on generated multi-channel images. In the experiments, we validated the proposed method with the images from SENSIAC (Military Sensing Information Analysis Centre) database and compared it with the state-of-the-art. The experimental results demonstrated the effectiveness of the proposed method on both accuracy and computational efficiency.*

### 1.0 INTRODUCTION

Automatic target detection (ATD) is key to automatic military operations and surveillance missions. In a military mission, sensors can be placed on the ground or mounted on unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) to acquire information. The acquired sensory data will be processed using ATD algorithms which aim to find bounding boxes where the targets can be located. Fast and accurate object detector can increase lethality and survivability of the weapon platform/soldiers. Whether the tactical scenario is the onslaught of an array of combat vehicles coming through the Fulda Gap, which was feared during the Cold War [1], or the identification of humans with intent to kill in an urban scene, the identification of the threat for avoidance and engagement is paramount to survival and threat neutralization.

Numerous ATD algorithms have been proposed during past decades. Generally, these algorithms can be classified into two main categories: 1) background modeling approaches, 2) and learning-based approaches.

Background modeling approaches assume that background pixels have a similar color (or intensity) over time in a fixed camera, and the background model is built on this assumption. The background is abstracted from the input image, and the foreground (moving objects) region is determined by marking the pixels in which a significant difference occurs. In [2], the authors modeled the background using a Kernel density estimation (KDE) method over a joint domain-range representation of image pixels. Multilayer codebook-based background subtraction (MCBS) model was proposed in [3], which can remove most of the non-stationary background and significantly increase the processing efficiency. Reference [4] proposed a motion detection

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

model based on probabilistic neural networks. Above methods are designated for the stationary camera scenario. In the work of [5] [6] [7], the authors proposed several schemes that can handle the problems in the moving camera scenario. The background modeling based methods are effective for detecting moving objects, whereas when the objects are still or moving slowly, those methods will always be unsatisfying.

Another popular category is the learning-based approaches. Traditionally, hand-engineered features like SIFT [8] or HOG [9] are firstly extracted and then fed into a classifier, such as Boosting [10], support vector machine (SVM) [11], and random forest [12]. The typical work in this paradigm is the deformable part models (DPM) [13]. More recently, Convolutional Neural Networks (CNN) shows its significant impact on ATD applications, which can achieve promising results in many difficult object detection challenges [14] [15] [16]. The *Overfeat* algorithm [17] successfully used CNN models in a sliding window fashion on ATD task. The algorithm has two CNNs: one for classifying if a window contains an object and the other for predicting the bounding box of the object. The most popular CNN-based ATD framework, *R-CNN*, was introduced in [18], which uses a pre-trained CNN to extract features from box proposals generated by selective search [19] and classifies the features with class specific linear SVMs. The significant advantage of this work is replacing hand-engineered features with features from the CNN feature extractor. Then, the variants of *R-CNN* were proposed to mainly solve the problem with computational load [20] [21] [22].



**Figure 1: Left: the appearance of target is undistinguishable from background environment. Right: the scale of target is various dramatically.**

Nevertheless, above mentioned ATD methods are only applicable to the general natural scene, and many challenges come up from the military scenario. First, the environment of battlefields is extremely complex. As shown in Fig.1, the appearance of the object includes color and texture is similar to the background in left example, because soldiers always attempt to decorate themselves or their vehicles similar to the environment in order to be invisible. And due to the vast battlefield, the scale of objects always dramatically changes with their distance to sensors. Thus, those environmental factors will always limit the ability of generic object detection algorithm. Second, military ATD application always runs on the embedded platform whose computational and memory resources are limited. In this case, the ability to run at high frame rates with relatively high accuracy becomes a crucial issue for military ATD.

Several image fusion based methods were proposed to enhance target representation in literature [23] [24] [25] [26]. Multiple images acquired with different range of electromagnetic spectrum were fused into one image by pixel-level image fusion algorithms such as PCA-based weighted fusion [26] and Discrete Wavelet Transform (DWT) [24], and then fed into an ATD system. When the fused images are used in ATD tasks, there are still deficient. To address the serious limitation, we propose a novel image fusion approach to improving detector performance in the military scenario, which exploits the significant advantage of the unsupervised feature learning characteristic of CNNs. Compared with high-level image fusion, the proposed method can achieve a higher accuracy and computational efficiency. In addition, we adopted the state-of-the-art generic object detection framework into the military scenario and used a cross-domain transfer learning

techniques to cover the shortage of insufficient data. In this way, the proposed framework achieved promising results on SENSIAC [34] military dataset. To sum up, our contributions in this work are as follows:

- Both spatial-temporal and multiple spectral bands information are employed in the ATD task.
- An unsupervised learning-based image fusion approach is proposed, which can automatically learn how to fuse the essential information from the different type of images into a set of discriminative feature maps.
- The proposed framework has the capability to transfer learning from visible images to our created complex fused images. Moreover, the proposed method can achieve 98.34% average precision and 98.90% top1 accuracy on SENSIAC datasets.

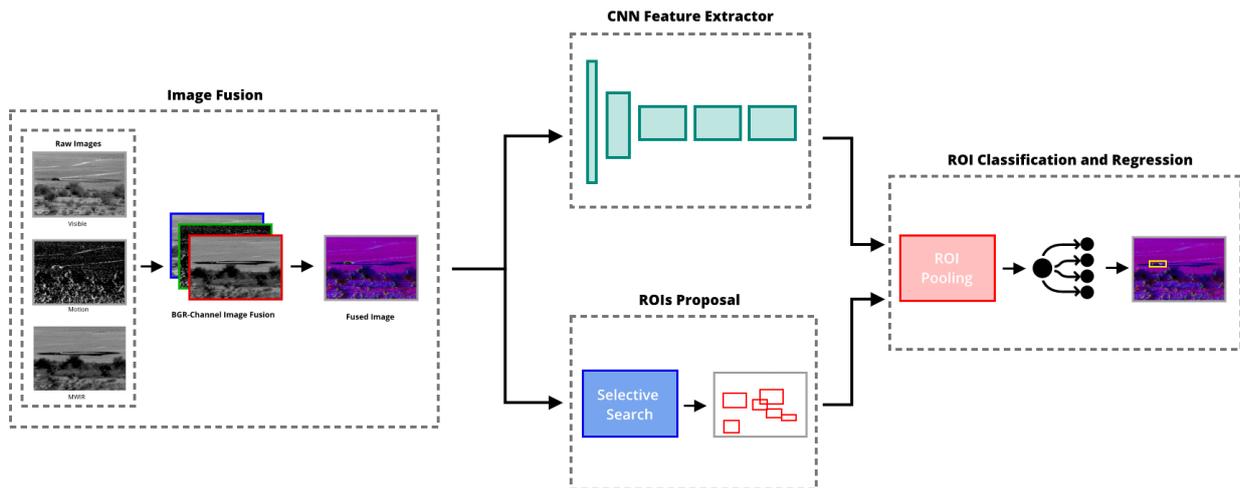


Figure 2: The pipeline of proposed object detector framework, which includes four main modules: 1) image fusion, 2) CNN feature extractor, 3) ROI proposal, and 4) ROI classification and regression.

## 2.0 METHODOLOGY

In this paper, we proposed a framework, namely, image fused object detector (IFOD). As illustrated in Fig.2, the whole system is composed of four modules: 1) an image fusion module, which can fuse three different type of images into a BGR image; 2) a CNN feature extractor, used for extracting high-level semantic representations from the fused image; 3) a region of interest (ROI) proposal module manipulated on fused image is utilized for generating hundreds or thousands of candidate bounding boxes, for each ROI on feature map produced by feature extractor module; and 4) an ROI classification and regression is performed to obtain fine bounding boxes and corresponding class.

### 2.1 Image Fusion

#### 2.1.1 Image Selection

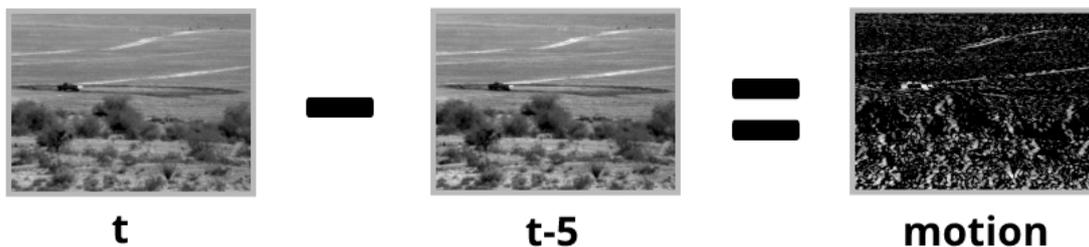
Multi-sensor data often provide complementary information for context enhancement, which may further enhance the performance of object detection. In our work, we investigated two type of images from different sensors, mid-wave infrared image (MWIR) and visible image (VI), respectively. In addition to the images acquired from these two sensors, we also incorporate motion image generated from two consecutive visible frames in order to complement sufficient description of objects.

**MWIR:** Depending on the different range of electromagnetic spectrum, the infrared (IR) spectrum can be divided into different spectral bands. Basically, the IR bands include the active IR band and the passive

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

(thermal) IR band. The main difference between active and passive infrared bands is that the passive IR image can be acquired without any external light source whereas the active IR required that. The passive (thermal) IR band is further divided into the mid-wave infrared (3-5  $\mu m$ ) and the long-wave infrared (7-14  $\mu m$ ). In general, the mid-wave infrared (MWIR) cameras can sense temperature variations over targets and background at a long distance, and produce thermograms in the form of 2D images. Its ability to present large contrasts between cool and hot surfaces is extremely useful for many computer vision tasks such as image segmentation and object detection. However, the MWIR sensor is not sensitive to cool background. And due to low-resolution sensor arrays and the possible absence of auto-focus lens capabilities, high-frequency content of the objects like edges and texture are mostly missed.

**VI:** The range of the electromagnetic spectrum of visible image is from 380  $nm$  to 750  $nm$ . This type of image can be easily and conveniently acquired via various kinds of general cameras. In comparison with MWIR image, the VI image is sensitive to illumination changes, preserve high-frequency information and can provide a relatively clear perspective of the environment. In most of the computer vision topics, the VI image has become major studying object for many decades. Thus, there are a large number of public VI datasets across many research areas. On the other hand, the significant drawbacks of VI image are that it has poor quality in the harsh environmental conditions with unfavourable lighting and pronounced shadows, and there is no dramatic contrast between background and foreground when the environment is extremely complicated such as the battlefield.



**Figure 3: The procedure of motion estimation: where  $t$  is the current frame and  $t-5$  is the previous 5th frame, and the motion is what our algorithm estimate.**

**Motion image:** In general, the moving objects are the targets in the battle fields. Therefore, estimating the motion of objects can provide significant cues to segment those targets. Various motion estimation algorithms have been proposed in recent decades, such as dense optical flow methods, points correspondence methods, and background subtraction. And each of them has shown effectiveness on many computer vision tasks. However, considering the trade-off between accuracy and computational complexity, we do not opt for any of the complicated motion estimation approaches but utilize a straightforward and easier to be implemented method. The method is illustrated in Fig.3, we estimate the motion map based on two consecutive frames. To be specific, the objective images are sampled at every 5 frames, and then force the current frame to subtract the last frame, the resulting image is the desired motion image. The method can be formulated as follow:

$$M_n(x, y) = |I_n(x, y) - I_{n-5}(x, y)| \quad (1)$$

where  $M_n(x, y)$  represents the motion value of frame  $n$  at pixel point  $(x, y)$  and  $I_n(x, y)$  denotes the pixel value of frame  $n$  at pixel point  $(x, y)$ .

In this way, we do not need to consider multiple frames to estimate background, like the background subtraction methods, and only the subtraction operator is employed in this procedure, which is more efficient

that other state-of-the-art methods. Even though this method can bring lots of noise in the motion image, this image still can provide enough complementary information in the later fusion stage.

### 2.1.2 Fusion Method

Here, we formalize the possible configurations of information fusion for object detection into three categories, namely, *pixel-*, *feature-* and *decision-level fusion* architecture. An illustration is shown in Fig. 4. Having these possibilities in mind will help to highlight the important benefits of our proposed fusion method in terms of efficiency and effectiveness.

**Pixel-level fusion:** A typical pixel-level fusion architecture is illustrated in Fig.4(a). Where it deals with the pixels obtained from the sensor directly and tries to improve the visual enhancement. Typically, multiple images from different sources are combined into one single image in a pixel-wise manner, after which it is fed into the object detection system to generate final results. One of the main advantages of the pixel-level fusion methods is the low computational complexity and easy implementation.

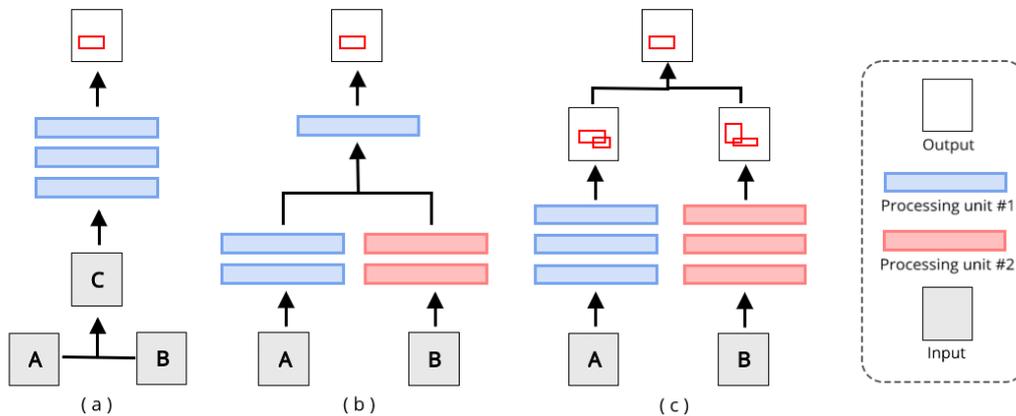


Figure 4: Illustration of different image fusion architectures: (a) pixel-level fusion architecture; (b) feature-level fusion architecture; (c) decision-level fusion architecture.

**Feature-level fusion:** As a higher-level fusion system to Fig.4 (a), one might pursue Fig.4(b), in which different type of images are simultaneously fed into their independent lower part of the entire object detection system, which is typically called feature extractor. For instance, this lower-level system might be the hand-engineered feature extractor for the traditional object detection system and high-level convolution layer for the CNN-based system. After which the concatenated features produced by the various independent lower system are fed into one upper (decision-making) system to produce the final results. Although this feature-level fusion is usually robust to the noise, it always requires almost double memory and computing resource to deal with feature fusion procedure in a parallel fashion, especially for the CNN-based methods.

**Decision-level fusion:** The decision-level fusion scheme, illustrated in Fig.4(c), operates on the highest level and refers to fusing discriminative results from different systems designed for various type images. Note that for an object detection system which usually based on machine learning algorithms, this high-level fusion probably could not establish a good relationship of interior characteristics between different type of images. In addition, it might also be practically challenging to be implement as this duplication would multiply the number of resources and running time.

In our framework, we proposed a novel image fusion approach which is similar to pixel-level fusion style. As you can see in the image fusion module in Fig. 2, firstly, the three type of raw images (MWIR, VI and Motion image) are concatenated into a BGR-style three-channel image where MWIR in the red channel,

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

---

motion image in green channel and VI in the blue channel. It is worthy noted that we do not modify any pixel values of the raw images but just put them into independent channels of final fused image. After the fused image is obtained, we fed it into a convolutional neural network (CNN) for training our object detector in an end-to-end manner. Meanwhile, the feature from different source images can be fused together in the last feature maps of the CNN in an unsupervised learning fashion. Therefore, compared with feature-level and decision-level fusion methods, our approach is easier to implement and low computational complexity. And to pixel-level fusion, we employ unsupervised learning style to fuse images from different sources instead of utilizing hand-engineered pixel-level fusion algorithms such as discrete wavelet transform (DWT) pixel-level image fusion methodologies.

### 2.2 Regions of Interest (ROIs) Proposal

As you can see in the ROIs proposal module in Fig.2, given an image, the ROIs proposal algorithms can output a set of class-independent locations that are likely to contain objects. Different from the exhaustive search "sliding window" paradigm which will propose every possible candidate locations and generate around  $10^4$ - $10^5$  windows per image, ROIs proposal methods try to reach high object recall with considerably fewer windows. In the most popular object detectors such as *R-CNN* [18] and *fast R-CNN* [20], they usually select *Selective Search* [19] method as their ROIs proposal module.

The *Selective Search* [19] is a ROIs proposal that combines the intuitions of bottom-up segmentation and exhaustive search. The whole algorithm can be simplified as follows. Firstly, [28] algorithm is adopted to create initial regions. Then the similarities between all neighbor regions are calculated and the two most similar regions are grouped together. After that, the new similarities are calculated between the resulting region and its neighbors. In this iterative manner, the process of grouping the most similar regions is repeated until the whole image becomes a single region. Finally, the object location boxes can be extracted from each region. Because of this hierarchical grouping process, the generated locations come from all scales.

### 2.3 Network Architecture

The great success of Convolutional Neural Networks (CNNs) in recent years aroused broader interest in CNNs-based generic object detection among researchers. In typically, a CNN comprises a stack of convolutional and pooling layers. The convolutional layer can generate feature maps by convolving the input feature maps or image with a set of learnable kernels. And the pooling layer can pool information of a given region on output feature maps in order to achieve down sampling and expansion of the receptive field.

The most typical CNNs-based object detector is the *R-CNN* [18], which utilize *Selective Search* method to generate a set of ROIs proposal from an input image and then feed each ROI to the CNN to obtain final results. However, due to lots of heavily overlapped ROIs need to go through the CNN separately, a large amount of redundant computation is consumed. *SPP-Net* [21] and *fast R-CNN* [20] successful solved this problem by proposing a Spatial Pyramid Pooling (SPP) and ROI pooling, respectively. They suggested the whole image can go through CNN once and the final decision is made at the last feature maps produced by the CNN by using their proposed pooling strategies.

Our proposed framework is illustrated in Fig.2. The fused three channel image is firstly fed into the CNN feature extractor to generate *conv* feature maps. It should be noted that the final *conv* feature maps in our project are also the fusing results of the three types of images by unsupervised learning. After which, for each ROIs generated by the ROIs proposal, we conduct an ROI pooling process directly on the *conv* feature maps instead of an input image to extract a fixed length feature vector. The reason to choose ROI pooling instead of SPP is that the gradients can propagate to the CNN layers in training stage and this can help CNN learn how to fuse the multiple channel-independent images in an unsupervised fashion. Finally, the extracted vector need to be sent to a fully connected neural network which has two output ports where one is for classification and another one is for bounding boxes regression.

Taking the trade-off between accuracy and computational complexity into account, the *VGGM* from [30] is selected as the CNN feature extractor in our framework. Specifically, The *VGGM* is a shallow version of *VGG16* [31] and wider version of *AlexNet* [32], but faster than *VGG16* as well as more accurate than *AlexNet*. More detail about the *VGGM* configuration can be seen in Table.1

**Table 1: Network configuration: The complete network architecture contains two modules, the first module is called CNN feature extractor which includes 5 convolutional layers (conv 1-5), the second module is the ROI classification and regression which has an ROI pooling layer and 4 fully connected layers.**

Name	Conv1	Norm1	Pool1	Conv2	Norm2	Pool2	Conv3	Conv4	Conv5	ROI Pooling	FC6	FC7
Input Channels	3	96	96	96	256	256	256	512	512	512	36	4096
Output Channels	96	96	96	256	256	256	512	512	512	36	4096	1024
Kernel Size	7*7		3*3	5*5		3*3	3*3	3*3	3*3	6*6		
Type	Conv	LRN	Max-pool	Conv	LRN	Max-pool	Conv	Conv	Conv	ROI-pool	Fc	Fc
Stride	2		2	2		2	1	1	1			
Pad				1			1	1	1			
Activation function	Relu			Relu			Relu	Relu	Relu			
Dropout											√	√

## 2.4 Training Details

### 2.4.1 Transfer Learning

Transferring general information between different data source for related tasks is an effective technique to help deal with insufficient training data and overfitting in the deep learning community. For instance, training a CNN model on the large ImageNet [14] dataset firstly and then fine tuning it on the domain-specific dataset. However, these general techniques are limited to transferring information between RGB (visible image) models.

The target training dataset includes the visible images, IR images, and generated motion maps, whose data distribution is significantly different to the large-scale public visible image datasets, such as ImageNet [14]. Our goal is to leverage the CNN model gain essential common knowledge from a large-scale visible dataset and then transfer these information for accelerating training in the domain-specific dataset as well as boosting overall performance.

Based on the general transfer learning techniques, the *VGGM* model is pre-trained on the large-scale RGB image dataset, ImageNet, which contains most common objects and scenes in daily life. Before training the network on the fused dataset, the weights of *conv1* to *conv5* in the network are initialized by transferring learned weights. Unlike some prior work, we do not freeze the lower layers of CNN and allow our network to adapt the new data distribution in an end-to-end learning setting.

### 2.4.2 Loss Function

As shown in Table.1, the network has two output heads. The first is for classifying each ROI, which will output a discrete probability distribution over two categories (background and target). And the second is for

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

regressing the bounding box offsets of ROI where for each category, it will output a tuple of  $(t_x, t_y, t_w, t_h)$ , the elements indicate the shift value relative to the central coordinate, height and width of original proposal ROI.

Similar to [20], the following negative log likelihood objective is used for classification:

$$L_{cls}(p, u) = -\log(p_u) \quad (2)$$

where  $p$  represents the predicted probability of one of the categories and  $u$  is the ground truth class.

For regression, the smooth  $L1$  loss function is used:

$$L_{bbox}(t^u, v) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L1}(t_i - v_i) \quad (3)$$

in which  $t^u$  is the bounding box offsets of the  $u$  class. And  $v$  is the true offsets.

In training stage, both of them will be combined together as follow:

$$L(p, u, t^u, v) = L_{cls}(p, u) + \lambda[u = 1]L_{bbox}(t, v) \quad (4)$$

$u = 1$  means only when the class is the target, the bounding box regression can be trained. The  $\lambda$  is used to control the balance between classification and regression, we set it as 1 in all experiments.

### 3.0 EXPERIMENTS

#### 3.1 Datasets

We evaluate our approach on a public released ATR database from Military Sensing Information Analysis Centre (SENSIAC) [34]. This database package contains 207 GB of MWIR imagery and 106 GB of visible imagery along with ground truth data. All imagery was taken using commercial cameras operating in the MWIR and visible bands. The types of targets are various, which include people, foreign military vehicles, and civilian vehicles. The datasets were collected during both daytime and night and the distance between cameras and targets varied from 500 to 5000 meters.

In our experiments, we only considered the objects of the vehicle and split 5 types of vehicles as training targets and 3 types of vehicles as testing targets. The name and appearance of the objects are showed in Fig.5. We selected each type of vehicles with 3 different range of distance between cameras and targets (1000, 1500 and 3000 meters). It should be noted that no matter how many fine-grained types of vehicle it has, we treated them as one class, "vehicle". Thus, the problem become a binary (vehicle and background) object detection problem. Moreover, because the format of raw data is video, we sampled the images at every 5 frames to maximize the difference between each frame. In total, we used 4573 images as training data and 2812 images as testing data.

#### 3.2 Experimental Setup

Our framework is implemented by using Caffe deep learning toolbox [33]. For the training machine, we used a computer with a NVIDIA GeForce GTX 1080 GPU, an Intel Core i7 CPU, and 32 GB RAM. For the setup of hyper-parameters, we followed the *fast R-CNN* [20]. Where we train all the networks each for 40000 iterations with initial learning rate 0.001 and 0.0001 for the last 10000 iterations, momentum 0.9 and weight decay 0.0005.



Figure 5: Appearance of targets in training data and testing data.

### 3.3 Evaluation

#### 3.3.1 Metrics

For all the metrics, we considered the detection as true or false positives based on whether the area of overlap with ground truth bounding boxes exceed  $0.5$ . The overlap area can be calculated by the below equation:

$$a_o = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} \quad (5)$$

where  $a_o$  denotes the overlap area,  $B_p$  and  $B_{gt}$  denote the predicted bounding box and ground truth bounding box, respectively.

**Mean Average Precision(mAP)** is a golden standard metric for evaluating the performance of an object detection algorithm, where it first calculates the average precision (AP) of each class and then average all the obtained AP values. Because there is only one class (vehicle) in our experiments, we select AP as one of the evaluation metrics. The AP value can be easily obtained by computing the area under the precision-recall curve.

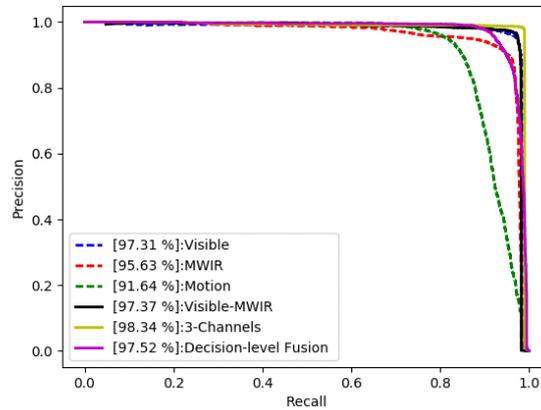
**Top1 Accuracy** is a metric that is widely used in classification tasks, where the probability of multiple classes is predicted and one having the highest score is selected, then the top1 precision score is computed as the numbers a predicted label matched the target label, divided by the number of whole data. In our case, there is only one target in each image. Thus, we can employ top1 precision metric in experiments to evaluate the performance of our framework in a practical scenario.

#### 3.3.2 Results and Analysis

Figure 6 shows the AP curves of the six incremental experiments. In independent image experiments, we can see that the CNN-based detector performed well enough in overall, especially for the single visible image which achieved 97.31% average precision and 98.04% top1 accuracy, as shown in accuracy column of Table.2. The visible-MWIR fused image get a better result than the best performance of single image. It should be noted that our 3-channels image achieved both the highest average precision (98.34%) and top1

## Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness

accuracy (98.90%) which means our method only false detect 16 frames in the totally 2812 testing frames. It is also interesting that even though the average precision of the decision fusion method is higher than the best single image method, but when it comes to practical application, its top1 accuracy is lower than the single visible image approach and it is extremely time-consuming in running time (3.961s).



**Figure 6: Average precision (AP) comparison between different experimental designs. Independent input of visible, MWIR and Motion image, fusion image of visible and MWIR image (Visible-MWIR), fusion image of visible, MWIR and Motion (3-Channels) and decision-level fusion, respectively.**

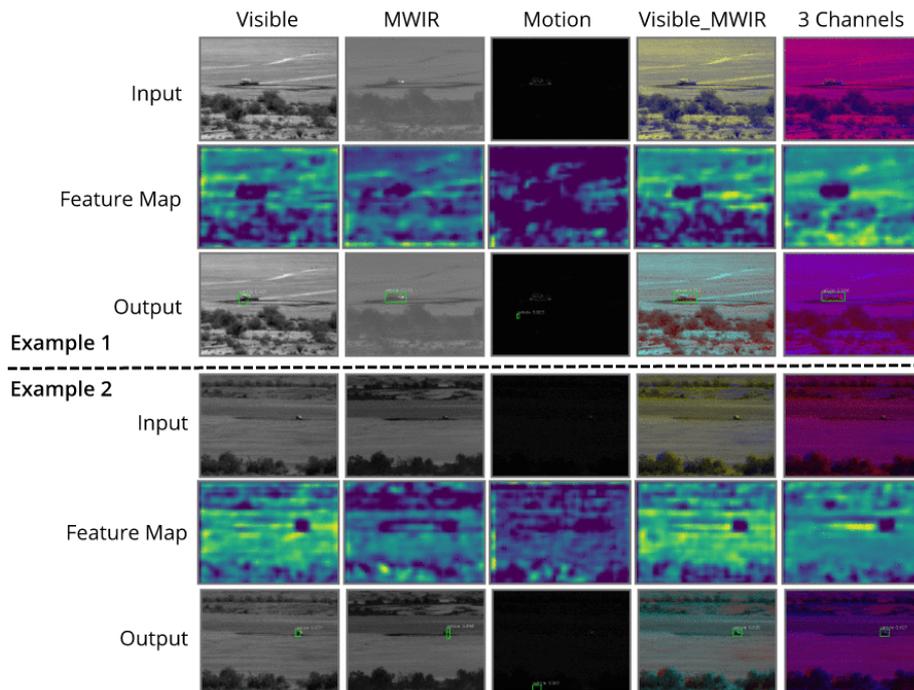
**Table 2: Performance comparison on accuracy and time cost of different methods.**

Methods	Accuracy (%)		Running Time ( <i>s/image</i> )		
	AP	Top1	ROIs Proposal	Networks	Overall
Visible	97.31	98.04	1.378	0.164	1.542
MWIR	95.63	96.91	<b>1.144</b>	0.069	1.213
Motion	91.64	92.39	1.167	<b>0.038</b>	<b>1.205</b>
Visible-MWIR	97.37	98.18	1.505	0.248	1.753
<b>3-Channels</b>	<b>98.34</b>	<b>98.90</b>	1.272	0.235	1.507
<b>Decision-level Fusion</b>	97.52	97.93	3.690	0.271	3.961

To further verify the effectiveness of our unsupervised image fusion method, we visualize the feature map of the last convolutional layer and the final output of our framework in Fig. 7. The feature map is the output of CNN feature extractor in Fig.2 and for the fused image, it is the fused high-level features. It could be reasoned that if the object in feature map is segmented clearly, the framework will get a better result. In the examples of Fig.7, we can see that the 3-channels can well fuse the complementary information from the three independent images and make its feature map get enhanced. And its final outputs also verify the fact that the enhanced feature map can boost the performance.

## 4.0 CONCLUSION

In this work, an unsupervised learning based image fusion method is proposed to integrate the ATD network, which fused visible, MWIR and motion information effectively. We further adopted state-of-the-art generic object detector for the battle field object detection. We also utilized cross-domain transfer learning techniques to deal with the insufficient data by training the model on large-scale visible image dataset firstly and then fine-tuning on the small-scale fused image dataset. The proposed framework was evaluated with the SENSIAC dataset. It achieved 98.34% average precision and 98.90% top1 accuracy. However, the processing time is still too long for real-time applications. This remains a topic for our future work.



**Figure 7: Example visualizing results of our framework. Example 1 and 2 demonstrate the performance of the different input types of the system on large and small object detection, respectively. Different columns denote different type of input image. The raw input image generated feature map and the final output are showed in consecutive rows. In the final output image, the green bounding box represents the position of object predicted by the system.**

## 6.0 REFERENCES

- [1] Ratches J A. Review of current aided/automatic target acquisition technology for military target acquisition tasks[J]. *Optical Engineering*, 2011, 50(7): 072001-072001-8.
- [2] Sheikh Y, Shah M. Bayesian object detection in dynamic scenes[C]//*Computer Vision and Pattern Recognition*, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005, 1: 74-7.
- [3] Guo J M, Hsia C H, Liu Y F, et al. Fast background subtraction based on a multilayer codebook model for moving object detection[J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2013, 23(10): 1809-1821.
- [4] Chen B H, Huang S C. Probabilistic neural networks based moving vehicles extraction algorithm for intelligent traffic surveillance systems[J]. *Information Sciences*, 2015, 299: 283-295.
- [5] Yun K, Lim J, Choi J Y. Scene conditional background update for moving object detection in a moving camera[J]. *Pattern Recognition Letters*, 2017, 88: 57-63.
- [6] Hu W C, Chen C H, Chen T Y, et al. Moving object detection and tracking from video captured by moving camera[J]. *Journal of Visual Communication and Image Representation*, 2015, 30: 164-180.
- [7] López-Rubio F J, López-Rubio E. Foreground detection for moving cameras with stochastic approximation[J]. *Pattern Recognition Letters*, 2015, 68: 161-168.
- [8] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. *International journal of computer vision*, 2004, 60(2): 91-110.
- [9] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//*Computer Vision and Pattern Recognition*, 2005. CVPR 2005. IEEE Computer Society Conference on. IEEE, 2005, 1: 886-893.

**Multi-Channel CNN-Based Object Detection for Enhanced Situation Awareness**

- [10] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]//Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. IEEE, 2001, 1: I-I.
- [11] Suykens J A K, Vandewalle J. Least squares support vector machine classifiers[J]. Neural processing letters, 1999, 9(3): 293-300.
- [12] Breiman L. Random forests[J]. Machine learning, 2001, 45(1): 5-32.
- [13] Felzenszwalb P F, Girshick R B, McAllester D, et al. Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010, 32(9): 1627-1645.
- [14] Russakovsky O, Deng J, Su H, et al. Imagenet large scale visual recognition challenge. International Journal of Computer Vision, 2015, 115(3): 211-252.
- [15] Everingham M, Van Gool L, Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.
- [16] Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European Conference on Computer Vision. Springer International Publishing, 2014: 740-755.
- [17] Sermanet P, Eigen D, Zhang X, et al. Overfeat: Integrated recognition, localization and detection using convolutional networks[J]. arXiv preprint arXiv:1312.6229, 2013.
- [18] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.
- [19] Uijlings J R R, Van De Sande K E A, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.
- [20] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE International Conference on Computer Vision. 2015: 1440-1448.
- [21] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[C]//European Conference on Computer Vision. Springer International Publishing, 2014: 346-361.
- [22] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//Advances in neural information processing systems. 2015: 91-99.
- [23] Bhatnagar G, Liu Z. A novel image fusion framework for night-vision navigation and surveillance[J]. Signal, Image and Video Processing, 2015, 9(1): 165-175.
- [24] Niu Y, Xu S, Wu L, et al. Airborne infrared and visible image fusion for target perception based on target region segmentation and discrete wavelet transform[J]. Mathematical Problems in Engineering, 2012, 2012.
- [25] Han J, Bhanu B. Fusion of color and infrared video for moving human detection[J]. Pattern Recognition, 2007, 40(6): 1771-1784.
- [26] Smeelen M A, Schwering P B W, Toet A, et al. Semi-hidden target recognition in gated viewer images fused with thermal IR images[J]. Information Fusion, 2014, 18: 131-147. C. L. Zitnick and P. Dollar, "Edge Boxes: Locating Object Proposals from Edges."
- [27] Felzenszwalb P F, Huttenlocher D P. Efficient graph-based image segmentation[J]. International journal of computer vision, 2004, 59(2): 167-181.
- [28] Dollár P, Zitnick C L. Structured forests for fast edge detection[C]//Proceedings of the IEEE International Conference on Computer Vision. 2013: 1841-1848.
- [29] Chatfield K, Simonyan K, Vedaldi A, et al. Return of the devil in the details: Delving deep into convolutional nets[J]. arXiv preprint arXiv:1405.3531, 2014.
- [30] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
- [31] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.
- [32] Jia Y, Shelhamer E, Donahue J, et al. Caffe: Convolutional architecture for fast feature embedding[C]//Proceedings of the 22nd ACM international conference on Multimedia. ACM, 2014: 675-678.
- [33] "Military Sensing Information Analysis Center (SENSIAC)," 2008, <https://www.sensiac.org/>.