Efficient Object Detection based on Deep Feature Fusion Network

Bo Zhao1,2*, Shuyun Liu2, Guizhong Liu1, Zhonglin Yang2, Zhiyang Ma2, Huini Fu2
1 Xi’an Jiaotong University, Xi’an, Shan Xi, 710049, China
2 China North Vehicle Research Institute, Beijing, 100072, China
*Corresponding author’s e-mail: whu_c2003@163.com

Abstract. Real-time object detection is crucial for many applications, such as automatic driving, security monitoring. It is vital for these application to perform accurate detection while keeping real-time performance. This paper propose an efficient object detection method based on deep feature fusing strategy. The feature extraction network employs deep convolution neural network to fuse multi-channel input, including colored image, infrared image and motion image. Multiple sources of images can provide complementary information, which is beneficial to accurate object detection. The backbone network is based on darknet-53 while integrating with feature aggression modules to capture features of shallow and deep activation maps. Our network is first trained on large-scale IMAGENET and COCO datasets, and then fine-tuned on small-scale datasets collected in real world. A series of quantitatively and qualitatively experiments are conducted to show superiority and efficiency of our method.

1. Introduction
Object detection is a fundamental and challenging computer vision task, which need to predict bounding box together with its property of each target in an image. Object detection has made tremendous advance in recent years thanks to the rapid development of deep convolutional neural networks (CNN). Generally, the object detection methods are categorized into two-stage detectors[1, 2, 3, 4] and one-stage detectors[5, 6, 7, 8, 9, 10]. Two-stage detectors are based on region proposal which perform classification and bounding box regression in separate stage while one-stage frameworks estimate object labels and location simultaneously. In general, two-stage detectors can achieve better precision while one-stage detectors usually run relatively faster. Current mainstream object detectors such as Faster R-CNN, SSD and YOLO depend on a set of pre-defined anchor boxes, which is vital for the success of detection. In recent years, anchor free methods[5, 13, 14] have played increasingly important role in object detection, which show excellent results compared to anchor-based ones.

The image classification methods have surpass human level on IMAGENET with the development of convolutional neural network (CNN). At the same time, general object detection achieve excellent results on several public datasets, such as PASCAL VOC[11], COCO[12]. However, these methods are not applicable for small target detection under complex scenarios, where it is difficult to distinguish between targets and their backgrounds. Under these backgrounds, the difference between target and its background is tiny, which is difficult to observe even for human eyes. Worse still, the object appearance and scales are changed during fast moving of interested target.
In this paper, we focus on object detection under complex background, where limited difference exists between targets and their background. Deep features are fused using multiple input channels, which includes RGB image, infrared image and motion image. The proposed method is relatively good for small object detection under complex scenarios.

2. Related Works
Deep convolutional neural network has been successfully adopted to object detection and achieve excellent results. Commonly used detection methods can be classified into two-stage detectors and one-stage detectors.

A two-stage detector [1, 2, 3, 4] first generates several region proposals, and then use bounding box regression while get target labels. R-CNN [1] employs CNN to extract target feature, and then send to classifier to perform classification. R-CNN had achieved amazing results compared to traditional hand-crafted method. R-CNN is very slow because of repeated CNN computation. Fast R-CNN [2] take whole image as input to compute activation map, which can prevent repeated computation to save cost. Faster R-CNN [3] employs a Region Proposal Networks (RPNs) that share convolutional layers with detection networks to further reduce computational cost, which show amazing accuracy and speed. FPN[4] integrate pyramid model into deep neural network to fuse top-down and bottom-up feature. In this way, high and low level feature are captured in detection procedure.

A one-stage detector [5, 6, 7, 8, 9] carry out object detection in single procedure to simultaneously get target labels and location. You Only Look Once (YOLO) [5, 8, 9] consider object detection as a regression problem to estimate bounding boxes and class labels in end-to-end manner, which can achieve real-time performance while obtain relatively good detection results. YOLO V3 modified backbone network and detection header to use multiple feature map, which can achieve state-of-art accuracy especially on smaller targets. SSD [6] generate confidence scores for each bounding box while make regression to better match object location. Li etc.[15] design a fresh feature aggregation modules to fuse shallow and deep convolutional features, which can get good performance especially for small targets.

Anchor-based detector taken anchor boxes as pre-defined sliding windows, similar to the traditional sliding-window approaches. On the contrary, anchor free detectors[5, 13, 14] do not depend on anchor boxes which significantly reduces the designed parameters and computational cost. YOLO V1 predicts bounding box at points near the center of targets, which can produce higher-quality detection. CornetNet[13] detects a pair of corners of a bounding box and then groups them to form final bounding box, which needs complicated post-processing to group the corner pairs. FCOS [14] employs all points in a ground truth bounding box to predict target location, which achieve state-of-the-art performance among one-stage detectors.

In this paper, we follow the idea of one-stage detection strategy, which is very accuracy while keeping real-time performance. In addition, we use feature aggregation modules and residual blocks to boost detection performance.

3. Multi-Channel Images Fusion
Generally, object detector takes RGB images as network input to perform feature extraction and detection. However, RGB image may have low quality especially in the dark environment. This paper employs a combination of RGB images, infrared images and motion images to get complementary information from different video source. In this way, detection performance can boost a lot especially for small object under complex background. Both CCD camera and thermal camera are used to form five-channel input. The two cameras are installed in a monitoring system, which is calibrated ahead of time to ensure the cameras always have same field of view.

**RGB images:** RGB images taken by CCD cameras can retain high-frequency information and keep sensitive to illumination changes. Under good lightening condition, RGB images can get sufficient information for target detection. Unfortunately, when light condition is not so ideal, images may suffer from poor quality, a lot of noise and low contrast.
**Infrared images:** Long-wave infrared cameras are used in our systems to form infrared images, which are sensitive to temperature discrimination over targets and their surroundings. When in darken environment, infrared images can provide supplement details for RGB images. However, infrared image may lose detail information of targets, such as edges and texture.

**Motion images:** For moving target, motion information can provide significant clue for detection. To estimate target motion, the n-th video frame is subtract with (n-3) th video frame to obtain motion images[15]. This can be formulated as follows:

\[ M_n(x, y) = I_n(x, y) - I_{n-3}(x, y) \]

where \( M_n(x, y) \) represents the motion image pixel value at point \((x, y)\) and \( I_n(x, y) \) denotes the pixel value of frame n at point \((x, y)\). This is a straightforward and easier method to achieve the trade-off between accuracy and computational complexity.

**Fusion Methods:** Typically, there are three types of image fusion strategy, including pixel-level, feature-level and decision-level, see Fig. 1 for illustration. Pixel-level fusion, as depicted in Fig. 1 (a), dealing with the raw pixels of multiple images directly to improve the visual enhancement. The combination is then fed into detection network to generate final results. Feature-level fusion, as depicted in Fig. 1 (b), send different source of images to separate feature extractor, which are then sent to upper layers to produce final results. This method needs double computation and storage resources. Decision-level fusion, as shown in Fig. 1 (c), refers to combine different results from separate detection network for different types of images. This strategy also require duplicated resources to maintain same inference time. Meanwhile, interior relationship between different types of images can not be well fused. This paper follows approach taken from Liu[15], which is similar to pixel-level strategy. Three types of images, RGB images, infrared images and motion images are employed to form five-channel input. The raw pixels are put into independent channel of fused images and then fed into neural network. The network can be trained in end-to-end manner. Our method is easy to implement but effective.

![Fig. 1 Different fusion architecture. (a) pixel-level fusion; (b) feature-level fusion; (c) decision-level fusion.](image-url)

### 4. Deep Feature Pyramid Detection Network

#### 4.1. Network Structure

Typically, a detection network includes a stack of convolutional, polling and full-connected layers, followed by proper activation functions. The convolutional layers are used to learn useful features by convolving input activations with filters. Pooling operation is used to down sampling the activation map to shrink the receptive field. The activation functions are used to add non-linearity of the network.

**Backbone.** We adopt an efficient one-stage detection strategy. The overall network structure is shown in Fig. 2. The backbone is a modified version of darknet-53[9] while integrating with feature aggregation modules. Darknet-53 includes 53 convolutional layers, and one average pool and full-connected layers. It is an efficient and accurate feature extractor which is much faster than ResNet-101 and ResNet-152 while having similar accuracy. In order to strengthen the feature of activation output,
Feature aggregation modules (FAM) are used to integrate shallow and deeper feature of convolutional layer. The details of FAM are depicted in section 4.2.

Feature Pyramid Structure. Darknet-53 is used to extract feature at several scales in bottom-up pathway. It builds a feature hierarchy consisting feature map at a scaling step of 2. These feature map has stride of \{2, 4, 8, 16, 32\} pixels with respect to input image. Following [18], feature pyramid structure is used to detect objects at different scales. One single scale image of different size is used to perform object detection while multiple levels of feature maps are produce at different scales.

The pyramid structure include bottom-up pathway and a top-down pathway. Bottom-up pathway is the commonly used feed forward convolutional computation, which perform feature extraction using various conv-pooling-activation steps. Typically, the output of each layer has strides of \{2, 4, 8, 16, 32\} pixels compared to input image. For our cases the network takes 800x800 images as input, the subsequent layers output the resolution of 400x400, 200x200, 100x100, 50x50, 25x25, respectively. The top-down pathway takes higher pyramid layers, which have low spatially resolution while keep rich semantic. These feature maps are further enhanced through bottom-up pathway via skip connections. Each skip connection integrate activation map of same spatial size from both bottom-up and top-down pathways. In this way, low-level and high-level features are fused, which is benefit for different size of object detection.

4.2. Feature Aggregation Block

Feature aggregation blocks, which are proposed by Li.[16], can integrate feature maps from different layers to form new feature channels. In this way, feature maps from shallow layers have larger receptive field while feature maps from deep layers often have more semantics information. The structure of FAM block is shown in Fig. 3 (a). Spatial Attention (SA) block is included in the FAM block to highlight target region, which is favor of detecting smaller objects.

SA blocks is shown in Fig. x(b). More details can be found in [16]. Given two input activation map with same channels: $X \in \mathbb{R}^{H \times W \times C}$ and $Y \in \mathbb{R}^{H \times W \times C}$, 1x1 convolutional layer is employed to integrate multiple input into one channel and then generate output.

![Fig. 2 The network architecture of our method.](image)
\[
O = F_{\text{conv}}(X) \in R^{H \times W}
\]

\(F_{\text{conv}}(\cdot)\) denote 1x1 CBR (convolutional + batch normalization + ReLU) operation. Then, an activation function is used after full-connected layer to get activation map, that is
\[
A = \sigma(WO) \in R^{H \times W}
\]

The final output \(Z\) is calculated by product of \(Y\) and \(A\),
\[
Z = Y \cdot S
\]

After a stack of transformation, image feature in shallow layers are effectively enhanced, which is beneficial for small object detection.

4.3. Data Augmentation
Data augmentation methods are used to improve the accuracy of detectors without hurt the inference cost. By using different kinds of data augmentation methods, the detection models have high robustness to targets under different backgrounds. We adjust brightness, contrast, hue, saturation, and noise of the images to achieve photometric distortions. We also add random scaling, cropping, flipping, and rotating to realize geometric distortion. In addition, Mosaic[20] is used to further enhance the detection ability of small objects. In addition, Mosaic can reduce overfit of the detection models. Mosaic mixes 4 training images into one, which allows detection of objects outside their normal context, seen Fig. 4.

4.4. Implementation Details
Network Input. This paper employs a pixel-level fusion method to integrate information from multiple image. The network input include five channels, which includes red, green, blue channels of CCD images, infrared image from thermal sensors and motion image. These images are combined to form five-channel input and fed it into convolutional neural network to train object detector. Consequently,
features from different sensors can be fused in pixel level in the internal of CNN. Similar to YOLO, the network can be trained in end-to-end manner. Compared with feature-level and decision-level fusion methods, our approach is easier to implement with low computational budget.

5. Experiments

Experimental setup: We use NVIDIA GeForce GTX 2080Ti GPU, an Intel Core i7 CPU with 64GB memory for training. All the codes are based on PyTorch for easy implementation. For the hyper-parameters setting up, we follow YOLO, where all the networks are trained 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.

Evaluation Metric. Mean Average Precision (mAP) is employed to evaluate performance of different network. Firstly, the average precision (AP) of each class is calculated and then AP values are averaged to get mAP. The AP value is computed using the precision-recall curve. Moreover, frame per-second (FPS) is used to evaluate the real-time performance of different methods.

Datasets. Because of limited datasets from real environment, we first train the network using large scale IMAGENET dataset and then retrain the network using images taken from binocular vision system to boost detection performance. Binocular vision system contain one colored CCD camera, one infrared camera. Motion image is obtained by subtraction of interval CCD images. There are four classes in the dataset, person, aeroplane, car and truck.

Comparison. The performance of our method and three other methods, Faster R-CNN, YOLO, Tiny-YOLO, are shown in Table 1. Our method achieves best detection performance while running in real-time. Faster R-CNN achieve relatively good mAP but very slow. Tiny-YOLO runs very fast but with low mAP.

More results: More detection results of our method are shown in Fig. 5. From the image, we can see low contrast between target and its background. Even hard for human eyes to discriminate. However, our method show preferably good results with sufficient train samples.

Table 1. Comparison of different methods

| Methods     | mAP  | person | aeroplane | car   | truck | FPS |
|-------------|------|--------|-----------|-------|-------|-----|
| Faster R-CNN| 82.7 | 77.3   | 84.2      | 82.9  | 86.4  | 7   |
| YOLO        | 83.2 | 79.1   | 83.8      | 83.1  | 86.8  | 45  |
| Tiny-YOLO   | 71.3 | 67.4   | 73.3      | 71.1  | 73.4  | 155 |
| Our methods | 84.9 | 81.2   | 85.3      | 84.5  | 88.6  | 50  |

Fig. 5. More detection results of our methods.
6. Conclusion
In this paper, we propose a novel detection structure that is based on ResNet. The network takes colored image, infrared image and motion image as input. Multiple image source can offer complementary information to boost detection accuracy. We also use feature aggregation modules to integrate information of different layers. The idea of transfer learning is employed to train the network using large-scale IMAGENET datasets first, and then fine-turn on small-scale datasets. Extensive experiments are conducted to show superior detection performance while running in real-time. In the future, we will collect more data under complex background to train more accurate detector that can be used in real application. Furthermore, we will seek more simple network structure to further accelerate inference speed while maintaining high accuracy.

Acknowledgments
This work was supported in part by Nature Science Foundation of China under Grant (61801439).

References
[1] R. Girshick, J. Donahue, T. Darrell, J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In CVPR, pp. 580-587 (2014).
[2] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In NIPS, pp. 91–99 (2015).
[3] J. Dai, Y. Li, K. He, J. Sun. R-FCN: object detection via region-based fully convolutional networks. In NIPS, pages 379-387 (2016).
[4] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, S. J. Belongie. Feature pyramid networks for object detection. In CVPR, pp. 936-944 (2017).
[5] J. Redmon, S. K. Divvala, R. B. Girshick, A. Farhadi. You only look once: Unified, real-time object detection. In CVPR, pp. 779-788 (2016).
[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, A. C. Berg. SSD: single shot multibox detector. In ECCV, pp. 21-37 (2016).
[7] C. Fu, W. Liu, A. Ranga, A. Tyagi, A. C. Berg. DSSD: Deconvolutional single shot detector. arXiv:1701.06659 (2017).
[8] J. Redmon, A. Farhadi. YOLO9000: Better, Faster, Stronger. arXiv:1612.0824225 (2016).
[9] J. Redmon, A. Farhadi. YOLOv3: An Incremental Improvement. (2018).
[10] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, A. Zisserman. The pascal visual object classes (VOC) challenge. In IJCV, 88(2), pp. 303–338 (2010).
[11] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick. Microsoft COCO: common objects in context. In ECCV, pp. 740-755 (2014).
[12] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In Proc. Eur. Conf. Comp. Vis., pages 734-750, 2018.
[13] Z. Tian, C. Shen, H. Chen, T. He. FCOS: Fully Convolutional One-Stage Object Detection. arXiv:1904.01355v5 [cs.CV] 20 Aug 2019.
[14] S. Liu, Z. Liu. Multi-Channel CNN-based Object Detection for Enhanced Situation Awareness. arXiv:1712.00075v1 (2017).
[15] W. Q. Li, G. Zh. Liu. A Single-shot Object Detector with Feature Aggregation and Enhancement. arXiv:1902.02923v1 (2019).
[16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR. pp. 770–778 (2016).
[17] T. Lin, P. Dollar, R. Girshick, K. He. B. Hariharan, S. Belongie. Feature Pyramid Networks for Object Detection. arXiv:1612.03144v2 [cs.CV] 19 Apr 2017.