This article has been accepted for publication in a future issue of IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

ComNet: Combinational Neural Network for Object Detection in UAV-Borne Thermal Images

Minglei Li, Xingke Zhao, Jiasong Li, and Liangliang Nan

***Abstract*— We propose a deep learning-based method for object detection in UAV-borne thermal images that have the capability of observing scenes in both day and night. Compared with visible images, thermal images have lower requirements for illumination conditions, but they typically have blurred edges and low contrast. Using a boundary-aware salient object detection network, we extract the saliency maps of the thermal images to improve the distinguishability. Thermal images are augmented with the corresponding saliency maps through channel replacement and pixel-level weighted fusion methods. Considering the limited computing power of UAV platforms, a lightweight combinational neural network ComNet is used as the core object detection method. The YOLOv3 model trained on the original images is used as a benchmark and compared with the proposed method. In the experiments, we analyze the detection performances of the ComNet models with different image fusion schemes. The experimental results show that the average precisions (APs) for pedestrian and vehicle detection have been improved by 2%∼5% compared with the benchmark without saliency map fusion and MobileNetv2. The detection speed is increased by over 50%, while the model size is reduced by 58%. The results demonstrate that the proposed method provides a compromise model, which has application potential in UAV-borne detection tasks.**

***Index Terms*— Combinational neural networks, model compression, saliency map, thermal image.**

I. Introduction

THE images obtained by unmanned aerial vehicles (UAVs) have great potential in a wide range of applications, such as traffic monitoring, urban security, and emergency response. Although image-based object detection techniques have been extensively studied in the past, most of the previous works focused on analyzing visible color images. One challenge of using visible images is that they are sensitive to lighting conditions, especially for images taken at night. Thermal images, on the contrary, require less illumination intensity, as they use infrared radiation emitted by objects. Hence, interests have been drawn toward exploring the use of thermal images to build intelligent systems for object detection in varying lighting environments [1]–[4].

Given the complementary nature between thermal images and visible images, researchers have explored constructing new architectures that fuse visible images and thermal images [5], [6]. However, generating registered visible–thermal image pairs is still an open problem. Besides, most UAV platforms have limited payload capacity and cannot load different imaging devices at the same time. Aerial visible images are susceptible to bad illumination, which makes object detection unreliable at certain conditions. In this work, we use only thermal images for object detection and focus on pedestrian and vehicle detection.

Compared with visible images, thermal images typically have blurred edges, low contrast, and high-levels of noise, which imposes challenges for object detection. When temperatures are very close in range, thermal imaging can lead to object confusion. Besides, UAV jitter might increase the level of blur at object boundaries in the images. To address these problems, we propose to augment thermal images with their boundary-aware saliency maps through a fusion strategy. The saliency map of an image represents a visual attention mechanism that highlights the pixel regions belonging to salient objects in a given scene [7]–[10]. However, the saliency maps discard texture information. Our work takes advantage of both saliency maps and thermal images to improve the performance of object detection. Especially, the proposed method uses a deep saliency network consisting of a prediction module and a refinement module to detect salient object regions with clear boundaries.

Our basic model is established by training state-of-the-art object detector YOLOv3 [11]. In YOLOv3, the detection is done by applying detection kernels on feature maps of three different sizes at three different places in the network. The model has a good performance for small objects by using short cut connections. Although the feature extraction capability of convolutional neural networks (CNNs) is continuously improved with the continuous deepening of the network layers, the model size and prediction speed bring many challenges in practical engineering. One the one hand, a deep CNN might contain dozens or even hundreds of layers, which leads to a large number of weight parameters. On the other hand, the large sizes of models result in high demands in device storage and computing resources. Thus, adjusting the structure of deep neural networks to achieve the best balance between accuracy and running time has become an urgent task.

Considering the limited computing capability of UAV platforms, we trained a combinational neural network "ComNet" model, which uses the lightweight network MobileNetv2 [12] to replace the traditional feature extraction module in YOLOv3. The proposed network can improve operation speed, and meanwhile, it also reduces the model size. Besides, we use the focal loss [13] to replace the original loss function in the YOLOv3 network to solve the number imbalance problem of negative samples. The experimental results show that, compared with the traditional networks, the object detection performance of the combined network improves in terms of detection speed and model size.

The main contributions of this article are as follows.

1) To the best of our knowledge, the first approach is augmenting UAV-borne thermal images with saliency maps to improve object detection performance. The method compensates for the defects of raw thermal images by extracting and infusing the boundary-aware saliency maps with thermal images.
2) Using ComNet, a balance between accuracy, network size, and speed. This deep model trained on the fusion images is lightweight and can be easily ported to a UAV-borne platform in the future.
3) A data set of thermal images containing original thermal images, pixel-level annotations for object detection, and corresponding saliency maps. The data set can be used for training and testing various deep learning techniques for object detection and segmentation. The data set is available at https://drive.google.com/drive/folders/1vCxXsKnK3dVB-bkT6XLbbQF7YTdS2CR0

II. Related Work

A. Object Detection

Traditional pattern recognition methods (e.g., support vector machines, conditional random fields, and maximum likelihood estimation) are still popular for objection detection for airborne data [14], [15], where supervised feature extraction is critical to the success of these methods. Recent advances in deep learning-based approaches have achieved state-of-the-art performances in various urban remote sensing tasks [16]–[18].

The first class of object detection deep networks is the Region-based CNN (R-CNN) series. The R-CNN approach [19] is first to get a manageable number of candidate object regions [20], [21] and evaluate convolutional networks independently on each region of interest (ROI). R-CNN was extended to allow attending to ROIs on feature maps using RoIPool, leading to fast speed and better accuracy. Faster R-CNN [22] advanced this stream by learning the attention mechanism with a region proposal network (RPN). Faster R-CNN is flexible and robust to many follow-up improvements and is the current leading framework in several benchmarks. On this basis, Mask R-CNN [23] extends Faster R-CNN by adding a branch for predicting an object mask in parallel with the existing branch for bounding box recognition.

On the other hand, the YOLO network [24], [25] takes the entire image in a single instance and predicts the bounding box coordinates and class probabilities for these boxes. The biggest advantage of using YOLO is its superb speed. YOLO also understands generalized object representation. The updated YOLOv3 network is a little bigger than the last version YOLOv2 but more accurate. It has shown a comparatively similar performance to the R-CNN algorithms. In our work, our task focuses on object detection and localization, rather than instance segmentation, so we use YOLOv3 as a baseline method for pedestrian and vehicle detection.

B. Saliency Detection

Different from other dense-labeling tasks, e.g., semantic segmentation and edge detection, the goal in salient object detection is to identify the visually distinctive regions or objects in an image and then extract the targets. Such processing is usually served as a preprocessing for further computer vision tasks.

Early research for saliency detection focused on handcrafted features and heuristic priors, e.g., center priors [26] and boundary background priors [27]. Along with the breakthrough of deep learning approaches, fully CNNs (FCNs) have been adopted for salient object detection [28], [29]. Zhang *et al.* [30] developed a reformulated dropout and a hybrid upsampling module with uncertain convolutional features (UCFs) to reduce the check-board artifacts of deconvolution operators as well as aggregating multilevel convolutional features in (Amulet) [31] for saliency detection, in which the saliency is mainly defined over the global contrast of the whole image rather than local or pixelwise features. To achieve accurate results, the methods must understand the semantic meaning of the whole image, as well as the detailed structures of the objects. Among these methods, U-shape-based structures [32], [33] receive the most attention because of their ability to construct enriched feature maps by building top–down pathways upon classification networks.

An unresolved problem is that linking features from different layers loses accuracy in recovering the boundaries. Although the features from deeper layers could help locate the target, the loss of spatial details might obstruct the features from shallower layers for recovering the object boundaries. A more proper way is to employ the multiscale features in a coarse-to-fine fashion and gradually predict the final saliency map. Considering that simply concatenating features from different scales may fail if disordered by the ambiguous information, coarse-to-fine solutions are employed in recent state-of-the-art methods, such as RefineNet [34], PiCANet [35], and RAS [36]. These methods address this limitation by introducing a recursive aggregation method that fuses the coarse features to generate higher solution semantic features stage-by-stage.

To get clear boundaries, BASNet [10] first makes a coarse saliency prediction with a deeply supervised structure, and then, it refines the residual of the saliency map with a bottom–up module and a top–down module. Boundary accuracy is improved by a hybrid loss implementation.

In this article, we adopt the BASNet structure to generate saliency maps from thermal images. The network learns the

transformation between the input image and the ground truth in a three-level hierarchy, namely, pixel-, patch- and map-levels. Using the hybrid fusing loss, the trained model can segment the salient object regions with clear boundaries. The enhanced boundary features can effectively compensate for the blurring characteristics of thermal images.

C. Model Compression and Acceleration

Most deep learning algorithms are computationally intensive and memory-intensive, making them difficult to deploy on embedded systems with limited hardware resources. To achieve the best balance between accuracy and runtime, many lightweight network structures have been proposed by adjusting the structure of networks. Compression without losing accuracy means that there is significant redundancy in the trained model, which shows the inadequacy of the current training methods.

SqueezeNet [37] introduces a fire module that consists of a squeeze layer and expand layers. The squeeze layer uses a 1×1 kernel to limit the input channel of the large kernel, so it effectively reduces model parameters and calculation costs. SqueezeNet can achieve a compression ratio of 50 times without losing model accuracy. ShuffleNet [38] makes full use of grouping convolution and channel shuffling to further improve the model efficiency. ShuffleNet solves the problem of information flow between groups while reducing the amount of calculation.

The MobileNet [39] and its variants [12], [40] are designed for lightweight mobile and embedded devices. MobileNetv1 uses depthwise separable convolutions to improve the computation efficiency. On this basis, MobileNetv2 adds a linear bottleneck and an inverted residual structure to form a more efficient basic module. MobileNetv3 uses a new nonlinear activation layer h-swish and a complementary network search method to search for a lightweight network. Compared with the previous versions, MobileNetv3 has the smallest size, but its structure is complex and is more difficult to train.

To tradeoff, our method uses MobileNetv2 to combine with the YOLOv3 model. Compared with the traditional YOLOv3 model, the proposed model runs faster and has a smaller model size, which can meet the requirement of lightweight platforms.

III. Method

A. Saliency Map Generation

We explore a deep network to extract saliency maps from thermal images. Since the backgrounds in UAV-derived images are complex, the saliency maps can highlight pedestrians and vehicles from the background. However, the saliency images discard all available texture information, which may cause missing detection. To address this problem, we fuse the thermal images with its corresponding saliency maps to improve the distinguishability of objects in the images. Since a grayscale raw thermal image is obtained by digitizing the thermal radiation, we convert the data to a pseudocolor image to fit the detector. As the first step, we map the grayscale image to a color image with R-G-B channels by the iron palette, where blue and purple are for slightly cold areas, and then, the higher temperatures are red, orange, and yellow. In the following steps, we use the RBG image for further processing.

Existing deep learning-based saliency detection methods focus on the accuracy of areas, instead of the quality of boundaries. As thermal images suffer from seriously blurred boundaries, we assume that the quality of the salient object boundaries has a large impact on the performance of object detection. Following the spirit of the UCF network [30] and BASNet [10], we use a boundary-aware method to generate saliency maps.

As shown in Fig. 1, the proposed architecture consists of a supervised prediction module and a residual refinement module. The prediction module is in charge of predicting a coarse map, and the residual refinement module is to refine the saliency map. The weights of the ResNet-34 model [41] are used to initialize the parameters of the feature extraction network.

The prediction module is a densely supervised encoder–decoder network incorporated with U-Net [32]. The encoder part extracts features from images, and a pooling method is used to obtain the high-level semantic features with progressively smaller resolution. The latter decoder part is responsible for the gradual reduction and amplification of high-level semantic information to gradually obtain the feature map with a large resolution. Subsequently, the decoder part outputs a coarse saliency map with the same size as the original one. The encoder and the decoder are directly connected, which add feature maps with the same resolution. This makes it possible for the final output feature map to take into account features from different levels.

The refinement module is designed as a residual block that refines the predicted coarse saliency maps S_{coarse} by learning the residuals S_{residual} between the saliency maps and the ground truth

$$S_{\text{refined}} = S_{\text{coarse}} + S_{\text{residual}}. \tag{1}$$

The output of this refinement module is the saliency map with better boundaries.

We adopt the loss function used in BASNet, which combines the loss of BINARY CROSS-ENTROPY (L_{bce}), structural similarity (SSIM, L_{ssim}), and Intersection-over-Union (L_{iou}). BASNet is deeply supervised with eight outputs, including seven outputs from the prediction model and one output from the refinement module. The loss is defined as

$$L = L_{\text{bce}} + L_{\text{ssim}} + L_{\text{iou}} \tag{2}$$

where L_{bce} corresponds to pixel-level supervision that is the most widely used loss in binary classification and segmentation

$$-\sum_{(r,c)}[G(r,c)\log(S(r,c)) + (1 - G(r,c))\log(1 - S(r,c))]$$

where $G(r,c) \in \{0,1\}$ is the ground-truth label of the pixel (r,c) and $S(r,c)$ is the predicted probability of being a salient object. It does not take into account the labels of the

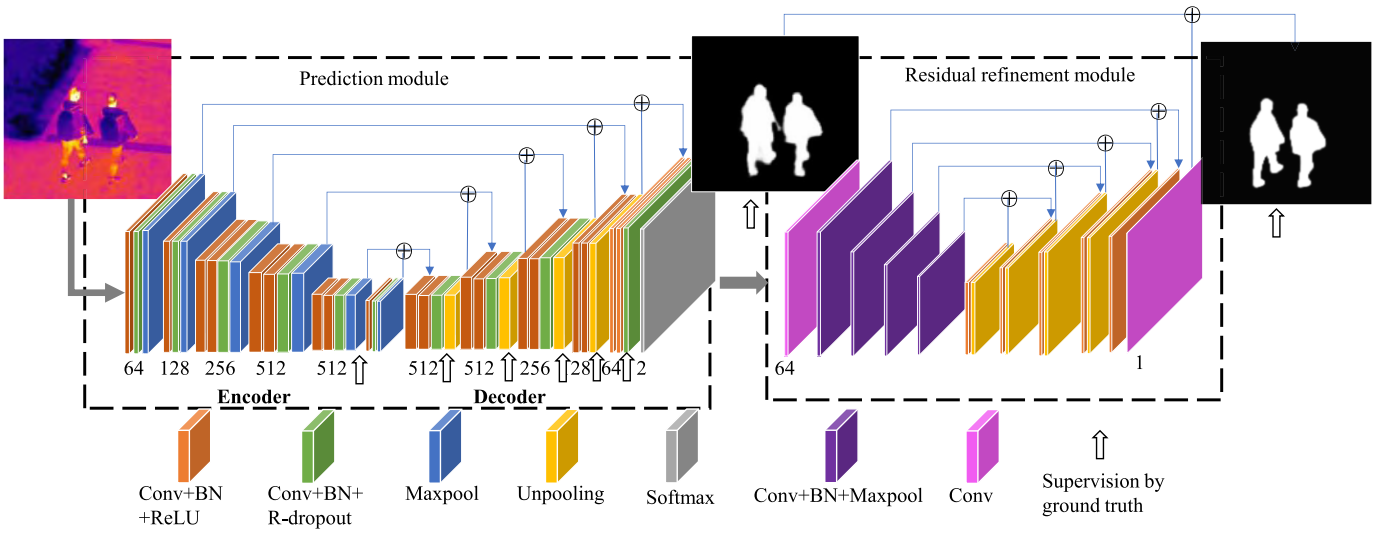

Fig. 1. Architecture of the boundary-aware saliency map detection network.

neighborhood. The weights of the foreground and background pixels are equal.

SSIM is originally proposed for image quality assessment. It captures the structural information of an image. Hence, BASNet integrated it into the training loss to learn the structural information of the salient object ground truth. L_{ssim} denotes the supervision at the patch level

$$\frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

where $x = \{x_j : j = 1, \ldots, N^2$ and $y = \{y_j : j = 1, \ldots, N^2$ are the pixel values of two corresponding patches (size $N \times N$) cropped from the predicted probability map S and the binary ground-truth mask G, respectively. (μ_x, μ_y) and (σ_x, σ_y) are the mean and standard deviations of x and y, and σ_{xy} is their covariance. $C_1 = 0.01^2$ and $C_2 = 0.03^2$ are used to avoid dividing by zero. L_{ssim} considers a local neighborhood of each pixel, and it assigns higher weights to the boundary, i.e., the loss is higher around the boundary.

L_{iou} denotes IoU loss, corresponding to the supervision at the level of the map

$$\frac{\sum_{r=1}^{H}\sum_{c=1}^{W} S(r,c)G(r,c)}{\sum_{r=1}^{H}\sum_{c=1}^{W} [S(r,c) + G(r,c) - S(r,c)G(r,c)]}$$

where $S(r, c)$ and $G(r, c)$ are consistent with those represented in L_{bce}.

When combining these three losses, BASNet uses BCE to maintain a smooth gradient for all pixels, while using IoU to give more focus on the foreground. SSIM encourages respecting the structure of the original image, by a larger loss near the boundary. Equipped with this hybrid loss, the architecture of salient object detection can effectively segment the object regions and accurately generate clear boundaries.

B. Fusion of Thermal Images With the Saliency Maps

After obtaining saliency maps, we test two different strategies for image fusion: 1) we replace one of the three channels of the thermal images with the saliency map to generate the fusion image and 2) we duplicate the saliency map three times and fuse them with the three channels of the thermal image by pixel-level addition, which uses an average weight ratio of 0.5. We expect that such a system would perform well especially in bad lighting conditions when objects are more indiscernible from their surroundings in thermal images.

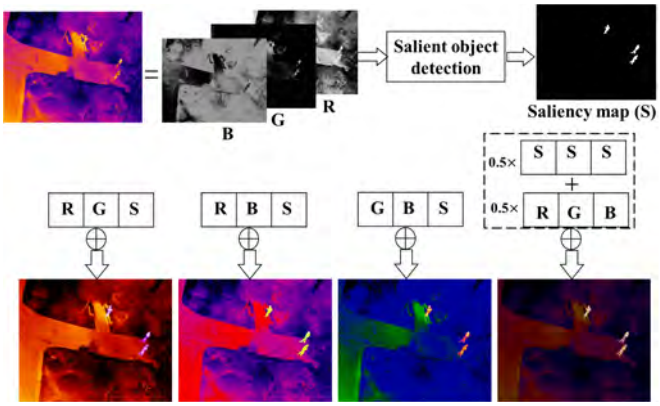

Fig. 2. Different methods for fusing the thermal image and the saliency map.

A brief view of the fusion schemes is shown in Fig. 2. We observed that the fusion images are capable of highlighting pixel regions of objects of interest, and meanwhile, they preserve texture information in the images. To test the performance of different fusion strategies, we train a series of object detection models on the following four types of data:

1) original thermal images R-G-B;
2) saliency maps of thermal images indicated by "S";
3) fusion images with channel replacement, namely, R-G-S, R-B-S, and B-G-S;
4) Fusion images with pixel-level weighted fusion, namely, 0.5S+0.5 (R-G-B).

C. Detection Models Based on YOLOv3 and Mask R-CNN

We use the network YOLOv3 [11] to complete the basic task of pedestrian and vehicle detection. The model trained on original thermal images is used as the evaluation benchmark.

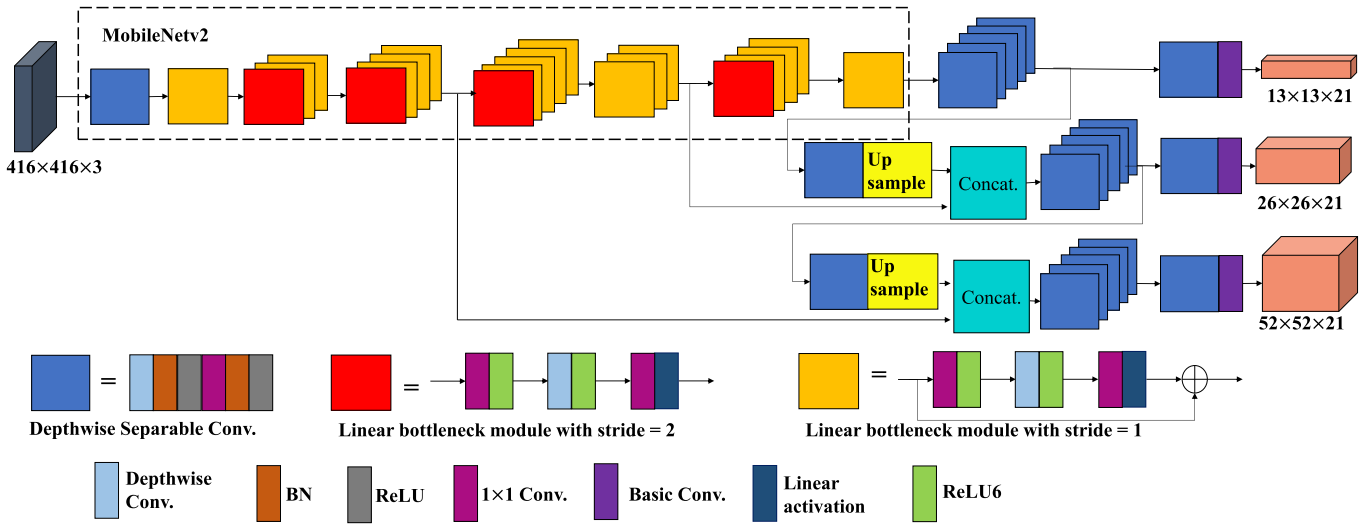

Fig. 3. Architecture of ComNet.

YOLOv3 divides an image into regularized grids. Each grid is only responsible for detecting the object whose center falls in the grid. Each grid needs to predict the boundary box (BBox) and category information in three scales. The method predicts BBox, object confidence, and category probability contained in all regions simultaneously using regression methods. As the network uses a multiscale detection structure, it has a good effect on the detection of small objects that are crucial for UAV-derived data.

We fine-tune the YOLOv3 model that is pretrained on the COCO data set [42] to generate our detection model. The benchmark method automatically starts training from DarkNet53 [43] rather than from scratch. It is equivalent to using the first few layers of the given model to extract shallow features and then falling into our classification at last.

The advantage of fine-tuning is that it does not need to retrain the model completely, thus improving efficiency. Generally, the accuracy of the new training model will gradually increase from very low values. The fine-tuning step enables us to quickly achieve high accuracy after a relatively small number of iterations. Using fine-tuning, better performance can be expected even if the data sets are small.

We also use Mask R-CNN to build a model that takes original thermal images as input. The pretrained weights for MS COCO [42] are used as a starting point to train our variation on the network. As our task is to classify individual objects from UAV-borne thermal images and localize them, using the bounding box can meet our needs. In this work, the Mask R-CNN model is used as a reference for algorithm comparison. The performance of the model is given in the following experimental part.

D. Model Compression Using MobileNetv2

To reduce model parameters and calculation costs, we propose a combinational network based on MobileNetv2 to slim YOLOv3 and call it ComNet. The architecture of ComNet is shown in Fig. 3.

YOLOv3 outputs three feature maps of different scales, where the lengths are 13, 26, and 52. This characteristic is one of the few improvements mentioned in the version v3 model: predictions across scales. Objects are detected in multiscales of different sizes. The finer the grid cell, the finer the objects can be detected. Using our different fusion images, the models are trained according to the object category (pedestrian and vehicle) and the size of the prior box.

Especially, we first remove the average pooling layer and the last convolutional layer in MobileNetv2. Then, the modified MobileNetv2 is used to replace the DarkNet53 in the YOLOv3 network. In MobileNetV2, there are two types of blocks. One is a residual block with a stride of 1. Another one is a block with a stride of 2 for downsizing. The ComNet retains the connection rule of DarkNet53 used in YOLOv3. That is, the last layers of the feature maps whose resolutions are of $8\times$ downsampling and $16\times$ downsampling of the input image are used as fine-grained features. These feature layers are fused with the high-level semantic features after upsampling in the detection network to enhance the object recognition ability of the network. Besides, the 3×3 convolution operation that occupies a large number of parameters in the network is replaced with depthwise convolutions.

There are two possible problems with the training data. First, the proportion of positive and negative samples is quite unbalanced. Especially, the number of positive samples is much smaller than that of negative samples. The second issue is that the gradient of the energy model is dominated by easily divided samples. Although the loss value of the easily divided samples is very low, they have a large number. Thus, they make a great contribution to the loss, resulting in poor convergence. Therefore, we use focal loss [13] to improve the original loss function as

$$Fl(p_t) = -k(1-p_t)^{\alpha}\log(p_t) \tag{3}$$

where p_t is the predicted probability and α and k are adjustable hyperparameters. In this article, we set $\beta = 0.5$ and $\alpha = 2$.

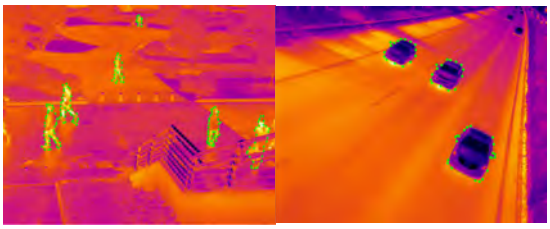

Fig. 4. Two thermal images from our data set with object annotated using *Labelme* [44] tool.

The purpose of the focal loss is to attenuate the intensity from the perspective of predictive confidence.

IV. Experiments

A. Data Sets and Experiment Setup

To train a deep detection model, a large number of samples are needed. To the best of our knowledge, there are few publicly available data sets of UAV-borne thermal images for detecting pedestrian and vehicle. Training the saliency map network from thermal images requires pixel-level annotations, which is a very tedious manual task.

In this work, we provide about 3000 thermal images with the corresponding annotations. A UAV was used to carry the thermal infrared camera to collect images at different periods. The images in the data set capture various scenes, such as the sports ground, expressway, campus, the entrance of the canteen with dense pedestrians at noon, and the corridor with sparse pedestrians at night. To ensure the validity of the model, the training set and the test set use the thermal images collected under different scenes. The training set and validation set are divided in the proportion of 80% and 20% during the network training. The original image size is 640×512. A total of 2434 training images are used including 3555 pedestrian instances and 3189 vehicle instances.

To generate the annotated data set, as shown in Fig. 4, we use the *Labelme* [44] tool to annotate the object pixels in images. The annotation here includes the borders of objects, categories, and the annotation for salient object detection. On the other hand, we created 541 images with similar annotations for testing our deep saliency detection network and the ComNet. The test data include 1213 pedestrian instances and 667 vehicle instances. The distribution of pedestrians and vehicles in each frame of the training set and test set is shown in Fig. 5. More than 75% of the images contain two to six instances.

We implement our network based on the publicly available framework: TensorFlow [45]. The deep models are trained and tested using an NVIDIA 1080ti GPU with 12-GB memory. Besides, our ComNet model has been transplanted to run on a personal laptop with an i5-8300 CPU with 4-GB memory and an NVIDIA Jetson Nano Board. The Nano Board has an integrated 128-core Tegra GPU, quad-core ARM A57 64-bit CPU, and 4-GB memory. The inference models have been tested on different machines.

To evaluate the detection models, we used the average precisions (APs) and frame per second (FPS) as the evaluation metrics for accuracy and speed, respectively. Besides, mean absolute error (MAE) and F-measure (F_β) are used to evaluate the performance of salient object detection. MAE computes the average absolute difference per pixel between predicted saliency maps and corresponding ground-truth annotations

$$\text{MAE} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} |\bar{S}(x, y) - \bar{G}(x, y)| \tag{4}$$

where W and H are the width and height of the image; $\bar{S}(x, y)$ and $\bar{G}(x, y)$ are the pixel values of the output saliency map and its corresponding ground truth. The F-measure F_β represents the weighted harmonic mean of precision and recall under the condition of a nonnegative weighted degree β. The higher F_β is, the better the model is. F_β is defined as

$$F_\beta = \frac{(1 + \beta^2)\text{Precision} \times \text{Recall}}{\beta^2 \text{Precision} + \text{Recall}} \tag{5}$$

where β^2 is set to 0.3 in our work. We set this value by learning from some existing literature [9], [46]. β^2 has an effect of raising the importance of precision. Thermal images, sometimes, have very confusing backgrounds. We do not want these regions to be detected as false positives and cause errors in subsequent target detections.

B. Salient Region Detection Results

We used thermal images to train the boundary-aware saliency network described in Section III-A. In the training phase, the size of each image in the training set is first adjusted to 256×256, and the training images are augmented by random flipping and cropping. The decoding network is trained from scratch with a learning rate of 0.01. The loss function converges after 60 000 iterations by using a batch size of 8. The entire training process took 7 h.

In the test phase, the input image size is also adjusted to 256×256 in the network to obtain predicted saliency maps. Then, the downsampled saliency map is reconstructed to the size of the original input image. Both adjustments are based on bilinear interpolation.

Fig. 6(a) shows two example thermal images of pedestrians and vehicles detected using our method. We tested the salient object detection model on original thermal images. The example saliency region masks generated by the proposed model are shown in Fig. 6(b). We observed that the distinguishability of vehicles is better than that of pedestrians in original thermal images, as there is a higher contrast between vehicles and the background road. The mask images do reflect the salient regions with clear boundaries. We also experimented with different strategies for generating fusion images. Especially, we replaced one of the RGB channels of the thermal images with the saliency maps, and another strategy was pixel-level weighted fusion. From Fig. 6(c)–(f), we can see that the combinations of saliency maps with thermal images all succeeded in illuminating the salient parts of the images while retaining the textural information in the images.

Referencing the manually annotated test data, the evaluation results of the salient object detection model show that F_β is 0.767 and c is 0.008. The inference speed of the saliency network is 39 FPS. The saliency detection and fusion of the

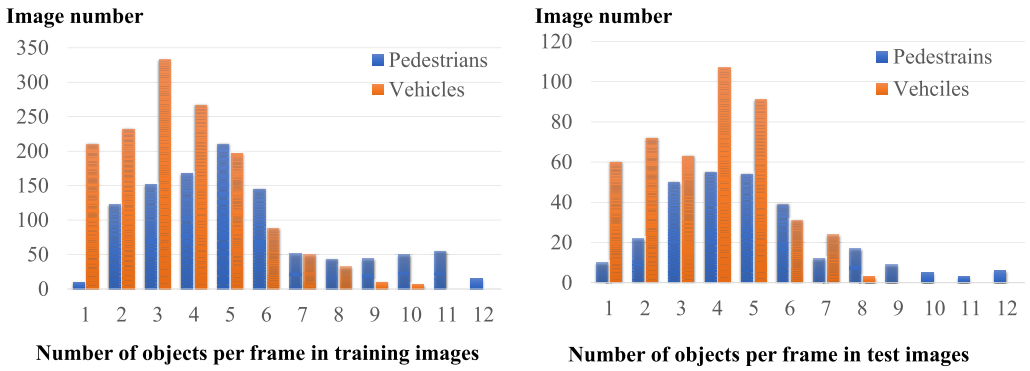

Fig. 5. Distribution of pedestrians and vehicles in the training images and test images.

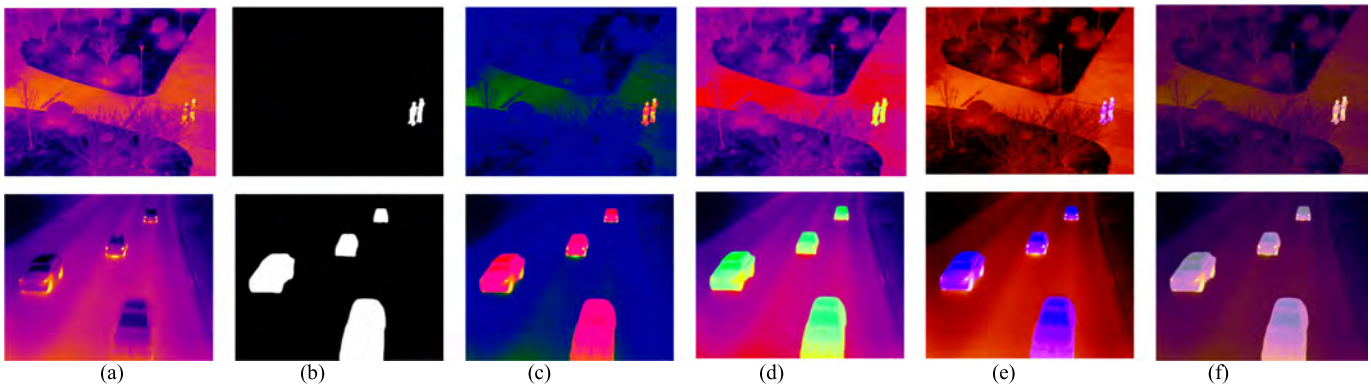

Fig. 6. Saliency map detection results and the fusion images. (a) Original thermal images. (b) Predicted Saliency maps. From (c)–(e), the fusion images generated by using saliency maps to replace R, G, and B channels of the thermal images. (f) Images generated by pixel-level weighted fusion.

images can be carried out in parallel, providing input data directly to ComNet.

To further illustrate the performance of the saliency detection method, Fig. 7 shows the qualitative comparison of the results with U-Net [32]. We can see that the U-Net result might merge adjacent people and distorts the shape of the objects. The salient regions that we detected have fine structures. Notice that the contours of annotations are polygons generated by *Labelme* [44] tool, with obviously broken line shapes. In contrast, the detected boundary is smoother and more consistent with the real shape of the object.

C. Object Detection Results of Different Methods

We trained the YOLOv3 model and the ComNet on different training images, i.e., on thermal images, saliency maps, and fusion images enhanced by different fusion methods. The image size is adjusted to 416×416 by bilinear interpolation. Both YOLOv3's and ComNet's backbone networks are pretrained on the Microsoft COCO data set [41], and fine-tuning with 100 epochs is conducted on the thermal images. The batch size is set to 8, and the initial learning rate is 0.001. We used Adam optimizer [47] to adaptively adjust the learning rate. The IoU threshold is set to 0.5, and the final prediction result is output after the nonmaximum suppression (NMS) operation.

After training models, the performance of detection results is evaluated in different settings using all methods: 1) thermal images + YOLOv3; 2) thermal images + ComNet; 3) saliency maps + ComNet; 4) replacing red channel of thermal images with saliency maps + ComNet; 5) replacing green channel of thermal images with saliency maps + ComNet; 6) replacing blue channel of thermal images with saliency maps + ComNet; and 7) weighted fusion of saliency maps and thermal images + ComNet. The inference performance of the methods on a GTX1080Ti machine is summarized in Table I. We observed that the saliency maps have an impact on the improvement of pedestrian and vehicle detection accuracy. Besides, the combination with MobileNetv2 improves the detection speed of the model.

1) Training YOLOv3 and Mask R-CNN: First, we trained YOLOv3 and Mask R-CNN on only the original thermal images. The YOLOv3 model size is 235 MB. The Mask R-CNN model size is 265 MB. Experimental results show that the APs of pedestrians and vehicles are 83.6% and 87.3%, respectively, for the YOLOv3 model, and the APs of pedestrians and vehicles are 87.1% and 91.5%, respectively, for the Mask R-CNN model. The YOLOv3 detection speed is 20 FPS, and the Mask R-CNN detection speed is 5 FPS. Mask R-CNN performs both object detection and instance segmentation at the same time, as shown in Fig. 8.

Though the detection accuracy of Mask R-CNN is higher than YOLOv3, its detection speed is far lower than YOLOv3. As our goal is to build applications where the efficiency of detection is an important factor, we choose YOLOv3 as the benchmark network. It can be seen from Fig. 8(c) that

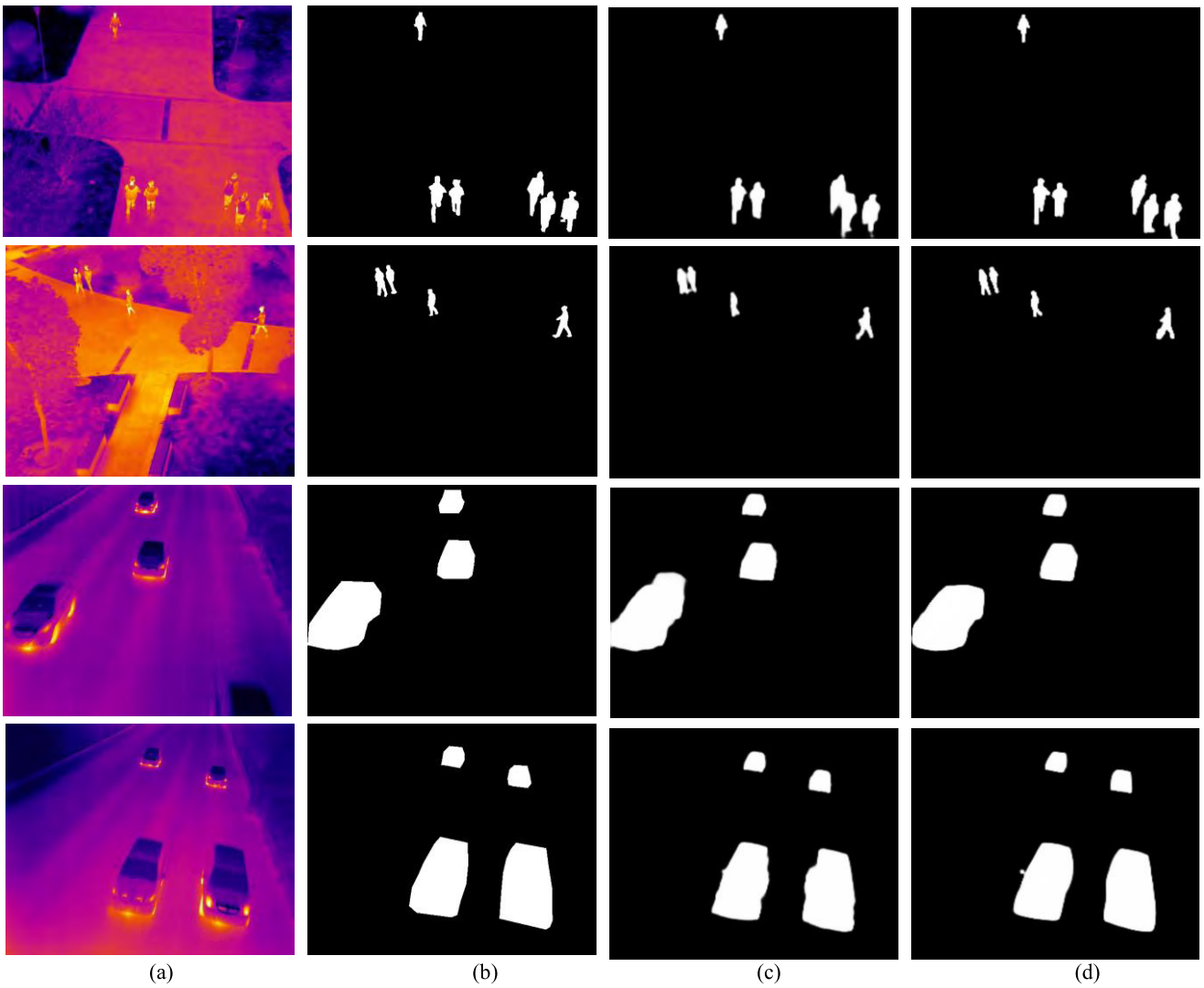

Fig. 7. Comparison of the proposed salient detection method with U-Net [32]. (a) Input images. (b) Manual annotations. (c) U-Net [32]. (d) Ours.

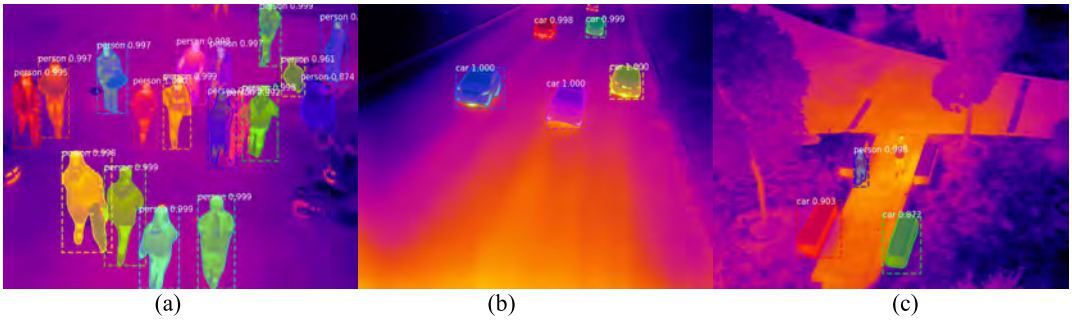

Fig. 8. Sample results of pedestrians and cars on Mask R-CNN. (a) Pedestrian detection. (b) Vehicle detection. (c) Incorrect results.

a pedestrian was missed, and the benches were incorrectly classified as vehicles. False positives indicate the necessity of using saliency maps to augment images to improve recognition performance. Given the comparison results, we agree that, for applications where efficiency is not a core consideration or the instance segmentation is advantageous, Mask R-CNN could be used for better accuracy.

b) Using only saliency maps: Using the model trained on only the saliency maps for detection, the APs are 77.1% and 82.0%, respectively. The results are 6.5% and 5.3% lower than that of the benchmark, and the detection speed is 21 FPS. We found that this method caused a large number of false detections and missing detections. Although the saliency map has certain application potential, as a binary image, it only

TABLE I
COMPARISON OF DIFFERENT TECHNIQUES (1080TI GPU)

		Pedestrian		Vehicle	
Data	Methods	AP	FPS	AP	FPS
Thermal R-G-B	YOLOv3	0.836	20	0.873	20
	ComNet	0.792	32	0.826	32
Saliency map S	YOLOv3	0.771	21	0.820	21
	ComNet	0.719	**34**	0.761	**34**
Fusion B-G-S	YOLOv3	0.927	20	0.932	20
	ComNet	0.880	32	0.889	32
Fusion R-B-S	YOLOv3	0.938	18	0.956	18
	ComNet	0.881	30	0.899	30
Fusion R-G-S	YOLOv3	0.905	19	0.972	19
	ComNet	0.857	31	0.925	31
Weighted fusion	YOLOv3	**0.944**	20	**0.978**	20
	ComNet	0.903	32	0.930	32

highlights salient regions in images and does not have any texture features. Therefore, the model using only the saliency map often fails.

c) Using the different fusion images: We have designed four methods to fuse thermal images with the saliency maps, by replacing one of the RGB channels of the thermal images with the saliency maps S and by weighted fusion. Experimental results show that after using saliency maps to replace the different channels of thermal images, the APs are higher than that of the benchmark, with increase rates between 5.9% and 10.2%.

Using the model trained on the weighted fusion images, the APs of pedestrian- and vehicle-detectors are 94.4% and 97.8%, which is an even better improvement of 10.8% and 10.5% over the benchmark, respectively. In general, the result of the model trained on the weighted fusion images is better than that of the model trained on the images replacing one channel of the thermal images.

These enhancements can be explained by the visual examples in Fig. 9. The fusion images highlight pedestrians and vehicles in the scene, enabling the detector to identify objects in a low-contrast context. It is also apparent to observe that the scheme using weighted fusion outperforms the scheme of the image channel replacement. That is because useful pixel information is lost during the image channel replacement process, and the replacement does not maintain the original structure of the image. The weighted fusion takes advantage of the complementary information from both thermal images and saliency maps, and thus, it is more discriminative.

Comparing the object detectors trained separately on thermal images, saliency maps, and the fusion images revealed that the saliency maps indeed contributed to improving performance. Since the input image size has not changed for all data, the detection speed in different schemes is almost the same as the benchmark, i.e., about 20 FPS.

2) Using ComNet as the Detection Network: Using the same data, the ComNet models show improvement in detection speed. Especially, the inference speed increased from 20 to 32 FPS on the GTX1080Ti machine, speeding up more than 50%. Meanwhile, the model size changed from 235 to 97 MB, decreased by 58%.

TABLE II
COMPARISON OF RUNNING TIME

Device	GPU / memory	YOLOv3 416×416 mAP = 0.94	SSD-MobileNetv2 480×272 mAP = 0.72	ComNet 320×320 mAP = 0.92
Workstation	GTX 1080Ti / 12GB	20 FPS	81 FPS	32 FPS
Laptop	Intel Core i5-8300 / 4GB	3 FPS	31 FPS	7 FPS
Jetson Nano	Tegra X1 / 4GB	2 FPS	27 FPS	6 FPS

It is worth noting that the AP values did not change much, while the size of the model has been reduced. Compared with the traditional YOLOv3 models, the detection accuracy of the ComNet model decreased by about 5% on the same data. More specifically, using the weighted fusion strategy, the APs of the proposed pedestrian- and vehicle-detectors are 90.3% and 93.0%. Compared with the YOLOv3 benchmark trained on original thermal images, the APs of the proposed combinational network model trained on fusion images increased by 6.7% and 5.7%, respectively. This indicates that the proposed neural network based on saliency map fusion images, indeed, improves the performance of objection detection.

The processing times of different models (YOLOv3, SSD-MobileNetv2, and ComNet) on different machines are compared. The mean AP (mAP) values for these models and running time are given in Table II. We see that the YOLOv3 model has the highest mAP but the lowest speed. The SSD-MobileNetv2 model is just the opposite. The ComNet model is faster than the Yolo model although the accuracy is slightly reduced. The current ComNet model cannot achieve real-time performance, but we can use the method of frame extraction to processing onboard. In terms of accuracy and efficiency, the proposed method has its advantages over other methods. It can be used as a reference for further research in this field.

As shown in Fig. 9, using saliency maps (c)-(I) and (c)-(IV) for augmentation is helpful to discover the missing pedestrians and vehicles in the original images (b)-(I) and (b)-(IV). In the saliency map (c)-(II), the detector mixes two overlapping pedestrians in the left area into one. Using the fusion images, the two pedestrians are correctly detected in (d)-(II), (e)-(II), (f)-(II), and (g)-(II). Similarly, using fusion images, the missing vehicle in (c)-(VI) is successfully found in (e)-(VI), (f)-(VI), and (g)-(VI). Note that the method of replacing one channel of the original thermal image with the saliency map may fail to detect the object, such as the small vehicle in (d)-(VI), indicating that the performance of channel fusion is worse than that of weighted fusion. This phenomenon is consistent with the statistics in Table I.

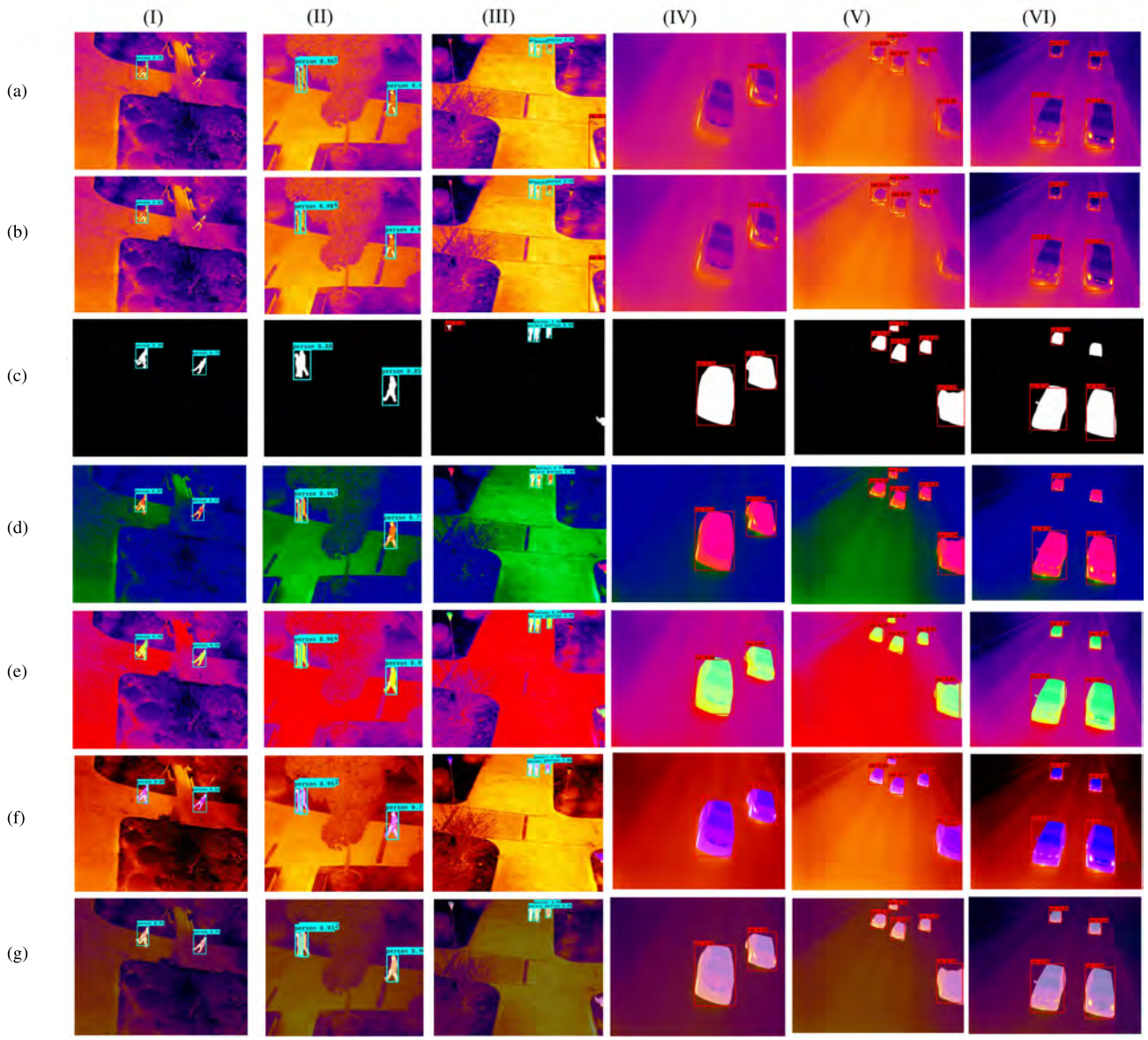

Fig. 9. Sample results from pedestrian detection on images and vehicle detection on image using different data and methods. (a) Thermal R-G-B + YOLOv3. (b) Thermal R-G-B + ComNet. (c) Saliency maps S + ComNet. (d) Fusion B-G-S + ComNet. (e) Fusion R-B-S + ComNet. (f) Fusion R-G-S + ComNet. (g) Pixel-level fusion + ComNet.

Fig. 9(b)-(III) shows an interesting example. Due to the similar shapes, the grass on the right was mistakenly detected as a vehicle. In Fig. 9(c)-(III), the street lamp on the upper left corner was detected as a vehicle because the temperature was higher than the surrounding environment. By combining thermal images with their corresponding saliency maps, the detector can eliminate these errors as seen in Fig. 9(d)-(III), (e)-(III), (f)-(III), and (g)-(III). The vehicle detected in Fig. 9(a)-(V) is missing in Fig. 9(b)-(V), showing the difference performance between YOLOv3 and ComNet.

The abovementioned results demonstrated the complementarity between thermal images and saliency maps, which confirms our expectation that the fusion of saliency maps can improve object detection accuracy. Besides, the combination of thermal images and saliency maps can be crucial to the detection performance of overlapping objects.

D. Influence of Data Source

The quality of the models is generally constrained by the quality of the training data. Thermal images captured from different devices do have different radiation sensing ranges, which may affect coloring results. As our images are captured by a single device, the characteristics of the images are similar. We have invented more images by randomly shifting and rotating existing images. Nevertheless, we know that increasing the richness of data sets from different devices can prevent the models from overfitting.

As we first use the iron palette to convert the grayscale thermal image to an RGB image, the converted color images will be different if the devices are different. Referring to [48], we assume that color has an impact on the test results but is not critical. To test the applicability of the models to different data, we retrieved some airborne thermal images from

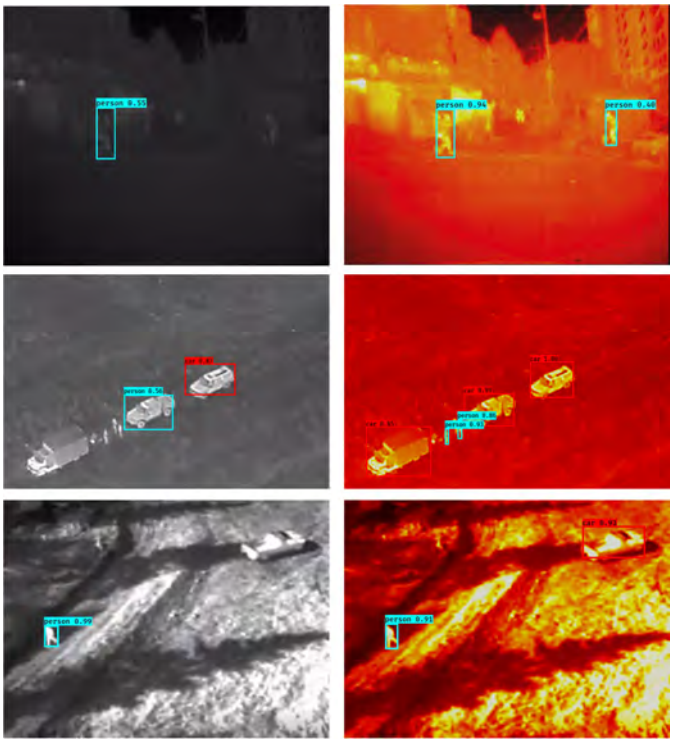

Fig. 10. Detection results on different images retrieved from the Internet. Left: detection results of the YOLOv3 model. Right: results of the ComNet model.

the Internet. These thermal images are captured by different devices with unknown conditions, and the ComNet model can get good detection results compared with the traditional YOLOv3 model, as shown in Fig. 10. A limitation lies in the lack of statistical analysis of large samples of different data types. We would enrich our data set continuously.

V. Conclusion

In this article, we use a UAV platform to collect the thermal images for object detection. To train and test the models, we prepared annotations for all images. Using deep learning, saliency maps of thermal images are extracted. Fusing thermal images with the extracted saliency maps, thermal images are augmented before they are put into the detection network. We compared different image fusion schemes, including channel replacement methods and the weighted fusion method. The fusion image provides complementary information for pedestrian and vehicle detections, which improves the performance of object detection. Besides, we proposed a combinational lightweight network ComNet, which is more efficient and has lightweight compared with the original YOLOv3 detection network. The experiments demonstrated that the ComNet model trained on fusion images has the potential for the detection of pedestrians and vehicles in the context of UAV-based applications.

References

[1] J. Portmann, S. Lynen, M. Chli, and R. Siegwart, "People detection and tracking from aerial thermal views," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 1794–1800.

[2] P. Wang and X. Bai, "Thermal infrared pedestrian segmentation based on conditional GAN," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 6007–6021, Dec. 2019, doi: 10.1109/TIP.2019.2924171.

[3] M. Li, L. Peng, Y. Chen, S. Huang, F. Qin, and Z. Peng, "Mask sparse representation based on semantic features for thermal infrared target tracking," *Remote Sens.*, vol. 11, no. 17, p. 1967, Aug. 2019, doi: 10.3390/rs11171967.

[4] Y. Sun, J. Yang, M. Li, and W. An, "Infrared small-faint target detection using non-i.i.d. Mixture of gaussians and flux density," *Remote Sens.*, vol. 11, no. 23, p. 2831, Nov. 2019, doi: 10.3390/rs11232831.

[5] D. Xu, W. Ouyang, E. Ricci, X. Wang, and N. Sebe, "Learning cross-modal deep representations for robust pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5363–5371.

[6] J. Liu, S. Zhang, S. Wang, and D. Metaxas, "Multispectral deep neural networks for pedestrian detection," in *Proc. Brit. Mach. Vis. Conf.*, 2016, pp. 1–13.

[7] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2214–2219.

[8] F. Zhang, B. Du, and L. Zhang, "Saliency-guided unsupervised feature learning for scene classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 4, pp. 2175–2184, Apr. 2015.

[9] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu, "Global contrast based salient region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 569–582, Mar. 2015, doi: 10.1109/TPAMI.2014.2345401.

[10] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7479–7489.

[11] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[12] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[13] T. Y. Lin, P. Goyal, R. Girshick, K. M. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 2999–3007, Aug. 2017, doi: 10.1109/TPAMI.2018.2858826.

[14] S. Cao, Y. Yu, H. Guan, D. Peng, and W. Yan, "Affine-function transformation-based object matching for vehicle detection from unmanned aerial vehicle imagery," *Remote Sens.*, vol. 11, no. 14, p. 1708, Jul. 2019, doi: 10.3390/rs11141708.

[15] J. Leitloff, S. Hinz, and U. Stilla, "Vehicle detection in very high resolution satellite images of city areas," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 7, pp. 2795–2806, Jul. 2010, doi: 10.1109/tgrs.2010.2043109.

[16] C. Heipke and F. Rottensteiner, "Deep learning for geometric and semantic tasks in photogrammetry and remote sensing," *Geo-Spatial Inf. Sci.*, vol. 23, no. 1, pp. 10–19, Jan. 2020, doi: 10.1080/10095020.2020.1718003.

[17] M. Coenen, F. Rottensteiner, and C. Heipke, "Precise vehicle reconstruction for autonomous driving applications," in *The ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Enschede, The Netherlands: Copernicus Publications, 2019, pp. 21–28.

[18] Y. Gong *et al.*, "Context-aware convolutional neural network for object detection in VHR remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 34–44, Jan. 2020.

[19] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.

[20] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 814–830, Apr. 2016.

[21] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *Int. J. Comput. Vis.*, vol. 104, no. 3, pp. 154–171, 2013.

[22] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017, doi: 10.1109/TPAMI.2016.2577031.

[23] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
[25] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.
[26] Z. Jiang and L. S. Davis, "Submodular salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2043–2052.
[27] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang, "Saliency detection via graph-based manifold ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3166–3173.
[28] G. Li and Y. Yu, "Deep contrast learning for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 478–487.
[29] S. S. S. Kruthiventi, V. Gudisa, J. H. Dholakiya, and R. V. Babu, "Saliency unified: A deep architecture for simultaneous eye fixation prediction and salient object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5781–5790.
[30] P. Zhang, D. Wang, H. Lu, H. Wang, and B. Yin, "Learning uncertain convolutional features for accurate saliency detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 212–221.
[31] P. Zhang, D. Wang, H. Lu, H. Wang, and X. Ruan, "Amulet: Aggregating multi-level convolutional features for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 202–211.
[32] O. Ronneberger, P. Fischer, and T. Brox, "UNet: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent*, 2015, pp. 234–241.
[33] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.
[34] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1925–1934.
[35] N. Liu, J. Han, and M.-H. Yang, "PiCANet: Learning pixel-wise contextual attention for saliency detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3089–3098.
[36] S. Chen, X. Tan, B. Wang, and X. Hu, "Reverse attention for salient object detection," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 234–250.
[37] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size," in *Proc. ICLR*, 2017, pp. 1–13.
[38] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," 2017, *arXiv:1707.01083*. [Online]. Available: http://arxiv.org/abs/1707.01083
[39] A. Howard *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*. [Online]. Available: http://arxiv.org/abs/1704.04861
[40] A. Howard *et al.*, "Searching for MobileNetV3," 2019, *arXiv:1905.02244*. [Online]. Available: http://arxiv.org/abs/1905.02244
[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[42] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Zurich, Switzerland, 2014, pp. 740–755.
[43] J. Redmon, *DarkNet: Open Source Neural Networks in C*. Accessed: 2019. [Online]. Available: https://pjreddie.com/darknet
[44] Kentaro Wada. (2016). *Labelme: Image Polygonal Annotation With Python Version: 4.2.9*. Accessed: Oct. 2, 2020. [Online]. Available: https://github.com/wkentaro/labelme
[45] *TensorFlow*. Accessed: 2019. [Online]. Available: https://www.tensorflow.org/federated
[46] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 733–740.
[47] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[48] B. Funt and L. Zhu, "Does colour really matter? Evaluation via object classification," in *Proc. 26th Color Imag. Conf.*, Vancouver, BC, Canada, 2018, pp. 268–271.