Multi-Scale Vehicle Detection in High-Resolution Aerial Images With Context Information

XIANGHUI LI, XINDE LI, (Senior Member, IEEE), AND HONG PAN, (Member, IEEE)
Key Laboratory of Measurement and Control of CSE, Ministry of Education, School of Automation, Southeast University, Nanjing 210096, China
Corresponding author: Xinde Li (xindeli@seu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 62073072, Grant 91748106, and Grant 61671151; and in part by the Advanced Research Project of the 13th Five-Year Plan under Grant 31511040301.

ABSTRACT Recently, unmanned aerial vehicles (UAV) are widely used in many fields due to the low cost and high flexibility. One of the most popular applications of UAV is vehicle detection in aerial images which plays an important role in traffic surveillance and urban planning. Although, many deep learning based detectors have achieved state-of-the-art (SOTA) performance in natural images, the significant variation in object scales caused by the altitude change of the UAV platform brings great challenges to these detectors for precise localization of vehicles in aerial images. To improve the detection performance for vehicles with different scales, we propose a novel detection algorithm which consists of three stages. In the first stage, to reduce the distortion of vehicles during image resizing and keep more information of aerial images, we utilize an image cropping strategy to divide the image into two patches. In the second stage, we combine the original image and two patches into a batch and detect vehicles with a Convolutional Neural Network (CNN). For feature representation in our detector, we propose Scale-specific Prediction to strengthen the multi-scale features of vehicles with context information. In the final stage, to fuse detections and suppress false alarms, we propose an Outlier-Aware Non-Maximum Suppression. Extensive experiments are conducted to demonstrate the superiority of the proposed algorithm by comparison with other SOTA solutions.

INDEX TERMS Vehicle detection, scale-specific prediction, single shot detector, high-resolution aerial images.

I. INTRODUCTION

The usage of UAVs is increasing rapidly recent years in plant protection [1], [2], traffic surveillance [3]–[5], disaster rescue [6], and urban planning [7]. One of the popular topics in the field of object detection in UAV images is vehicle detection [8]–[12]. The distribution of vehicles in different districts can provide essential information for traffic supervision and urban planning. However, aerial images may be taken at different altitudes, thus making vehicles present at different scales in different images as shown in Figure 1. For images captured from high-altitude UAVs, the ground sampling distance (GSD) is usually higher than 0.1 meter and the size of a standard vehicle is typically with $48 \times 16$ pixels [7]. For images captured from low-altitude UAVs, the GSD may reach the centimeter level and the typical size of a vehicle is $180 \times 80$ pixels [13]. Therefore, learning an effective representation of vehicles with different scales is an essential challenge for vehicle detection in aerial images.

Recently, deep-learning based detection models have achieved the SOTA performance on generic object detection in natural images [14]. There are two main frameworks for detection tasks: two-stage detectors and one-stage detectors. Regions with CNN features (RCNN) [15] and its variants [16]–[19] are typical two-stage detectors which first propose potential object regions, followed by bounding box regression and object classification. Whereas, YOLO (You Only Look Once) [20]–[22] and SSD (Single Shot Multibox Detector) [23] are typical one-stage detectors which embed region proposal and object classification in an end-to-end framework. However, these SOTA detectors may not achieve the same performance on UAV images due to their inherently differences from natural images. Furthermore, the vehicles are usually arbitrarily distributed in aerial images which
makes it difficult for these detectors to cover vehicles and results in IoU (intersection over union) distribution imbalance problem [24]. In addition, there are many small vehicles in UAV images [25] which makes it necessary to leverage more details of small vehicles in high resolution UAV images.

To tackle the problems in vehicle detection in high-resolution UAV images, we propose a novel algorithm which mainly consists of three steps. The framework of our approach is shown in Figure 2. First, we utilize an adaptive image cropping strategy for images with aspect ratio deviating from 1. However, some vehicles may be divided into two or more parts. Therefore, we conduct detection on both the original image and the image slices. To strengthen the feature representation of multi-scale vehicles, we propose scale-specific prediction modules. After detecting on these images, there are many redundancies which have to be removed. Meanwhile, there are many false alarms while some of them are with extreme scales or aspect ratios. We consider these false alarms as outliers and propose OA-NMS to remove them. The main contribution of our work is as follows:

1) First, we analyse the IoU distribution imbalance problem in vehicle detection in high resolution aerial images and propose a novel adaptive framework to utilize more information of aerial images.

2) Second, we analyse the difficulty in feature representation of multi-scale vehicles in high resolution aerial images and propose a novel scale-specific prediction based single shot detector (SSP-SSD) to include more context information to strengthen the feature representation of multi-scale vehicles.

3) Third, we propose a novel Outlier-Aware Non-Maximum Suppression (OA-NMS) to fuse results and remove false detections.

The paper is organized as follows: in section II, we briefly review the related works of vehicle detection in UAV images. In section III, we elaborate the proposed approach. In section IV, some experiments are conducted to verify the performance of our method by comparing with some other SOTA models. Finally, we conclude our work in section V.

II. RELATED WORKS

A. TRADITIONAL VEHICLE DETECTION

Traditional vehicle detection algorithms are mainly based on handcrafted descriptors including Local Binary Pattern (LBP), Harr-like features, Scale-Invariant Feature Transform (SIFT) and Histogram of Oriented Gradient (HOG) [13], [26], [27]. Kembhavi et al. [7] combined color probability maps of vehicles with surroundings, structural characteristics of objects called pairs of pixels and HOG features to represent vehicles. Zhou et al. [28] utilized local steering kernel features to describe vehicles and applied a sliding window strategy for vehicle detection in arbitrary orientations, which is very time-consuming. Moranduzzo and Melagni [29] detected vehicles in UAV images with different descriptors. They observed that the integration of the original SIFT features with color and morphological features had the best performance. A combination of diverse features including HOG, LBP and opponent histograms was proposed in [30] to detect cars in aerial images. Chen Ziyi [8] et al. proposed a multi-order feature descriptor which contains color, texture and high-order context information to describe vehicles. However, the background of aerial images is usually very complex, and parts of other objects (e.g., building roof, road marks, trash bins and electrical units) may appear similar to vehicles in some circumstances. These factors make it
difficult for handcrafted descriptors to completely distinguish vehicles from the background.

**B. DEEP LEARNING BASED VEHICLE DETECTION**

With the development of deep learning, many CNN based algorithms have been proposed to strengthen feature representation of vehicles in UAV images. Audebert et al. [31] proposed a segment-before-detect algorithm to detect vehicles in very high resolution (VHR) UAV images. They utilized SegNet [32] to obtain a pixel-level semantic map followed by a CNN to classify instances. However, the semantic segmentation would cost a lot of time and lead to low efficiency. Tao et al. [33] proposed a similar framework and added additional post-processing step to improve the recall rate which is more time-consuming. Ammour et al. [9] segmented the image into small homogeneous superpixels for the prediction of vehicle candidate locations. They extracted image features with a pre-trained CNN network and applied a linear support vector machine (SVM) for region classification. This algorithm can not effectively deal with the scale change of vehicles. When each vehicle was completely segmented into a single superpixel, it could achieve good detection results. However, if the vehicles are so small that many vehicles are in one superpixel or the vehicles are so large that one vehicle is divided into many superpixels, it would be difficult to achieve good results. Lars Sommer et al. [34] modified the region proposal module of Faster RCNN (FRCNN) [17] according to the characteristics of UAV images. Yohei Koga et al. [35] utilized a hybrid deep CNN for vehicle detection and proposed a hard example mining algorithm to select the most informative samples for training. Li et al. [11] developed a unified framework for simultaneous vehicle detection and counting. They employed a scale-adaptive anchor generator and a feature extraction module to maximize the mutual information between vehicle classes and features. These algorithms utilized the feature pyramid to construct multi-scale feature representation for vehicles at different scales. In the feature pyramid, low-level features were used to represent small objects while high-level features were used to represent large objects. However, for small vehicles, low-level features...
can hardly provide enough information, which makes context information important for object representation. In addition, a large receptive field may introduce unnecessary background information, which inevitably degrades the feature learning performance for small vehicles [36]. Therefore, we propose SSP-SSD and utilize context information to strengthen the feature representation of multi-scale vehicles.

In addition, to keep more details of small vehicles in high resolution UAV images, some cropping methods have been proposed [25], [37], [38]. Ünel et al. [25] simply cropped the images into many pieces and conducted detection on both the original image and the image slices. George et al. [37] proposed a selective tiling method to select the slices which may contain objects. Kouris et al. [38] proposed a density map prediction method for guidance of image cropping. These methods aim at achieving higher accuracy with sacrifice in time efficiency. With the increasing of the slice number, the method will be more and more time-consuming. In addition, these methods adopt NMS [39] to fuse results. However, except redundant detections, there are many false alarms. Kouris et al. [40] investigated the relation between the vehicle scales and UAV altitude and utilized it to remove false positives. However, this algorithm needs additional information e.g. the UAV altitude when taking the image and it should assume that the image looks at a flat area, which is not always the case for surfaces with different levels of elevation and changing UAV heights.

Therefore, to achieve the balance between the accuracy and time efficiency, we propose this simple but efficient image preprocessing strategy. In addition, we propose an effective OA-NMS to remove redundancies and false detections which does not need extra information.

III. METHODOLOGY

The framework of the proposed algorithm is illustrated in Figure 2. Our framework can be divided into three parts: Image Preprocessing (IP), Scale-specific Prediction and OA-NMS. In this section, we will describe these parts in details.

A. IMAGE PREPROCESSING

As shown in Figure 3, image scaling may result in vehicle distortion as is shown in Figure 3. For one-stage detector, the pre-defined anchors are utilized to cover vehicle candidate regions. Normally, IoU [41] is used to determine the overlap between an anchor $bb_i^m$ and the ground truth $bb_j^s$, which is defined in Eq.(1).

$$
IoU = \frac{area(bb_i^m \cap bb_j^s)}{area(bb_i^m \cup bb_j^s)} 
$$

where $area(bb_i^m \cap bb_j^s)$ denotes the intersection area between the anchor and the ground truth while $area(bb_i^m \cup bb_j^s)$ represents their union.

If the IoU is larger than a predefined threshold, the region covered by the anchor box $bb_i^m$ can be selected as a vehicle candidate region. Therefore, the design of anchors with predefined scales and aspect ratios has a great influence on the detection performance. However, when a UAV image is scaled to fit the input size of the detector, the aspect ratios of vehicles would also undergo changes, as illustrated in Figure 3. For instance, the bus in the original image has an aspect ratio of 48 : 155 which approximately equals to 1 : 3.23. However, the aspect ratio of the bus becomes 1 : 5.74 after image scaling. Therefore, to fully cover these vehicles, we have to define a large number of extra anchors, while most of them are false candidates and may result in false alarms. Additionally, due to the arbitrarily distribution of vehicles, the pre-defined anchors have low IoU with a large number of ground truth. These two factors make the IoU distribution skewed towards lower IoUs and cause IoU distribution imbalance problem [24].

Therefore, to reduce the IoU distribution imbalance problem and keep more information, we propose an adaptive image preprocessing method in Figure 2. We define $H_i$ and $W_i$ as the height and width of the original image respectively. We utilize $S_h$ and $S_w$ to represent the height and width of image slices respectively. We calculate $S_h$ and $S_w$ according to Algorithm 1. To save computation, we limit the number of image slices to a maximum of two. Whereas, if the $S_w$ or $S_h$ is simply assigned with $W_i/2$ or $H_i/2$ respectively, some small vehicles may be divided into two parts and difficult to be detected in the original images. To solve this problem, we add $\delta$ to $S_w$ or $S_h$ to ensure that small vehicles remain intact in at least one slice as shown in Figure 2. For vehicles in 4K images, $\delta$ has little influence on the aspect ratio of vehicles in slice. For instance, after image cropping, the aspect ratio of the aforementioned bus is approximately equal to 1 : 3.55 after scaling if $\xi$ equals 0.1. Therefore we adopt a uniform cropping strategy for all images.

B. SCALE-SPECIFIC PREDICTION

After image preprocessing, we apply a CNN to detect vehicles in the original image and image slices. The detection network is built on SSD and the architecture of the network is demonstrated in Figure 2. We utilize Resnet101 [42] as
and 3

tion. Therefore, we add additional branches to include more
features which do not contain sufficient semantic informa-
levels of features. For small vehicles, we utilize low-level
SSP for vehicles with different scales.

generate missing and false detections. Therefore, we propose
to the lack of context information, low-level features often
through regression, as shown in Figure 4 (a). However, due
to the lack of context information, low-level features often
generate missing and false detections. Therefore, we propose
SSP for vehicles with different scales.

First, we utilize feature pyramid to construct different
levels of features. For small vehicles, we utilize low-level
features which do not contain sufficient semantic informa-
tion. Therefore, we add additional branches to include more
context information as shown in Figure 4 (a). However, due
to the lack of context information, low-level features often
generate missing and false detections. Therefore, we propose
SSP for vehicles with different scales.

Algorithm 1 Image Cropping Algorithm

Input: $W_I, H_I$
Output: $S_w, S_h$
1: $\eta = \text{round}(\max(H_I, W_I)/\min(H_I, W_I))$
2: $\delta = \min(H_I, W_I) \times \xi, \xi \in [0, 0.2]$
3: if $W_I \leq H_I$ then
4: $S_w = W_I$
5: $S_h = \max(H_I/\min(\eta, 2), W_I) + \delta$
6: else
7: $S_w = \max(W_I/\min(\eta, 2), H_I) + \delta$
8: $S_h = H_I$
9: end if

backbone for feature extraction. The original prediction layer
of SSD obtains the object’s category and location directly
through regression, as shown in Figure 4 (a). However, due
to the lack of context information, low-level features often
generate missing and false detections. Therefore, we propose
SSP for vehicles with different scales.

![Diagram](image.png)

**FIGURE 4.** Prediction modules: (a) original prediction module; (b) prediction module for small objects; (c) prediction module for large objects.

features of an anchor, $y_i$ and $bb^an_i$ are the corresponding label
and coordinates of the anchor. We utilize $m_{i,j} = [0, 1]$ to
indicate the matching of the $i$th anchor to the $j$th ground truth.
During the training process, if the $i$th anchor has the highest
IoU which is larger than 0.5 with the $j$th ground truth box
$bb^g_j$, we consider the anchor as positive and assign the label
$y_i = 1$ and $m_{i,j} = 1$. Meanwhile, it is appended to $A_+$ set.
If an anchor’s IoU with all ground truth boxes is lower than
0.2, it is predicted as negative and then appended to $A_-$ set.
We select the $3M$ negative anchors with highest probability
into the training set. Based on the above definitions, the loss
function can be defined as:

$$
\text{Loss}_n(x, m, \hat{bb}^{pr}, bb^g) = \frac{1}{M} (\text{L}_{cls}(p(x), m)) + \alpha \text{L}_{loc}(m, \hat{bb}^{pr}, bb^g) \tag{2}
$$

where $M$ is the number of matched anchors, if $M = 0$, the
loss is set to 0, $\hat{bb}^{pr} = \{\hat{b}_x, \hat{b}_y, \hat{w}^y, \hat{b}_h\}$ denotes the offset
between an anchor and the corresponding predicted box. The
classification loss $\text{L}_{cls}$ is the cross entropy loss [44] and is
defined as follows:

$$
\text{L}_{cls}(x, m) = - \sum_{i \in A_+} m_{ij} \log(p(x_i)) - \sum_{k \in A_-} \log(p(x_k)) \tag{3}
$$

where $p(x_i)$ represents the probability of an anchor belonging
to vehicle while $p(x_k)$ denotes the probability of the anchor
belonging to backgrounds.

The localization loss $\text{L}_{loc}$ is a smooth L1 loss [16] between the
predicted box and the ground truth. The localization loss is
defined as follows:

$$
\text{L}_{loc}(m, \hat{bb}^{pr}, bb^g) = \sum_{i \in A_+} \sum_{k \in [x, y, w, h]} m_{ij} \text{smooth}_{L1}(\hat{b}_i - \hat{b}_j^k) \tag{4}
$$

$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2 & |x| < 1 \\ |x| - 0.5 & |x| \geq 1 \end{cases} \tag{5}$

where $\{\hat{b}_x^i, \hat{b}_y^i, \hat{w}^y_i, \hat{b}_h^i\}$ is the offset between a ground
truth and the corresponding anchor $bb^an_i = \{b_x^i, b_y^i, w^y_i, h_i\}$,
i.e., $\tilde{b}_{jx} = (b_{jx} - l_{j})/b_{i}^{w}$, $\tilde{b}_{jy} = (b_{jy} - l_{j})/b_{i}^{y}$, $\tilde{b}_{jw} = \log(b_{jw}^{w}/b_{i}^{w})$, $\tilde{b}_{jh} = \log(b_{jh}^{h}/b_{i}^{h})$.

We utilize stochastic gradient descent (SGD) to optimize the detection network in our training process. In the inference stage, a UAV image and its slices are fed into the network for vehicle localization.

C. OUTLIER-AWARE NMS

After detecting vehicles on both original images and image slices, there are many redundant detections, which makes it necessary to merge all detections belonging to the same vehicle into a single detection. The most popular methods are NMS [39] and SoftNMS [45]. NMS starts with the detections $BB$ with scores $S$ and adopts Eq. (6) to re-score the detections. First, it selects the detection with highest score $\hat{b}$. After removing it from the set $BB$, NMS appends it to the final detection set $D$ and removes any results which have IoU scores greater than a threshold $N_T$ from the set $BB$. This process is conducted repeatedly until the set $BB$ is empty. However, NMS sets a hard threshold when re-scoring boxes, which means that if two true positives have an overlap greater than $N_T$, one with a lower score is deleted. Therefore, to solve this problem, Soft-NMS utilizes a Gaussian penalty function as a re-scoring function, in Eq.(7), to assign the box $\hat{b}$ with a lower score if it has a high overlap with $\hat{b}_{j}$.

$$s_j = \begin{cases} s_j, & \text{IoU}(\hat{b}_{j}, \hat{b}) < N_T \\ 0, & \text{IoU}(\hat{b}_{j}, \hat{b}) \geq N_T \end{cases}$$

(6)

(7)

However, in aerial images, parts of other objects which appear similar to vehicles would also be mistaken as positive samples. For example, in Figure 5 we display that some false alarms with extreme scales or aspect ratio have low scores compared with true positives in the same image. The area ratio $AR_v$ of a vehicle $v$ in image $I$ is defined in Eq.(9).

$$AR_v = \frac{\text{area}(v)}{\text{area}(I)}$$

(9)

where area$(v)$ denotes the area of vehicle $v$, area$(I)$ denotes the area of image $I$ which contains $v$.

Therefore, we consider these false alarms as outliers and propose an OA-NMS method to remove them. We propose an outlier-aware re-scoring function defined in Eq.(10). In OA-NMS, we utilize $\hat{\varphi}$ and $\hat{\xi}$ to represent the average area and aspect ratio of bounding boxes in one image. In different images, $\hat{\varphi}$ and $\hat{\xi}$ have different values.

$$s_j = \varphi_j(3s_j + f(\gamma_j/\hat{\varphi}, \alpha) + f(\xi_j/\hat{\xi}, \beta))/5, \forall \hat{b}_{j} \notin D$$

(10)

$$f(x, \eta) = e^{-(\ln(x))^2/\eta}$$

(11)

$$\hat{\varphi} = \frac{1}{k} \sum_{j=1}^{k} \gamma_j, \hat{\xi} = \frac{1}{k} \sum_{j=1}^{k} \xi_j$$

(12)

where $\xi_j$ and $\gamma_j$ are the area and aspect ratio of the $j$th box respectively.

$$\frac{\partial f(x, \eta)}{\partial x} = -2 \frac{\ln x}{\eta x} e^{-(\ln(x))^2/\eta}$$

(13)

The penalty function is continuous and Eq.(13) gives its derivative. It can be seen that when $x > 1$, $\partial f(x, \eta)/\partial x < 0$ and when $x \in (0, 1)$, $\partial f(x, \eta)/\partial x > 0$. As shown in Figure 6, the penalty function gets maximum value at $x = 1$ and there would be a high penalty when $x$ deviates far from 1. Therefore, if a detection has an extreme size or aspect ratio compared with other detections in the same image, its $f(\gamma/\hat{\varphi}, \alpha)$ or $f(\xi/\hat{\xi}, \beta)$ should have a small value and thus will be removed. However, according to Eq.(10), if a detection has a high score larger than $5N_T$, it can be considered as a true positive and will not be removed. Algorithm 2 gives the pseudo code of the OA-NMS algorithm.

IV. EXPERIMENTS

In this section, we conduct several experiments to demonstrate the effectiveness of our algorithm on our captured dataset and UAVDT [46]. We utilized DJI Matrice 100 (M100) platform to obtain images for our dataset. All the experiments are conducted on a desktop computer with Intel
Algorithm 2 OA-NMS Algorithm

Input: $BB = \{\hat{bb}_{re}^1, \cdots, \hat{bb}_{re}^k\}$, $S = \{s_1, \cdots, s_k\}$, $AE = \{\xi_1, \cdots, \xi_k\}$, $AR = \{\gamma_1, \cdots, \gamma_k\}$, $N_t$

Output: $D, S$

1: $D = \emptyset, \bar{\xi} = \frac{1}{k} \sum_{j=1}^{k} \xi_j, \bar{\gamma} = \frac{1}{k} \sum_{j=1}^{k} \gamma_j$
2: while $BB \neq \emptyset$ do
3: \hspace{1em} $hs = \text{argmax}(S)$
4: \hspace{1em} $D = D \cup \hat{bb}_{re}^{hs}$, $BB = BB - \hat{bb}_{re}^{hs}$, $S = S - s_{hs}$
5: \hspace{1em} for $\hat{bb}_{re}^{j}$ in $BB$ do
6: \hspace{2em} $s_j = \vartheta_j(3s_j + f(\xi_j/\bar{\xi}) + f(\gamma_j/\bar{\gamma}))/5$
7: \hspace{2em} if $s_j \leq N_t$ then
8: \hspace{3em} $BB = BB - \hat{bb}_{re}^{j}$, $S = S - s_j$
9: \hspace{2em} end if
10: \hspace{1em} end for
11: end while

Core i7 4790k CPU, 32 GB memory and a NVIDIA Titan X GPU (with 12 GB graphics memory). The operating system is windows 7.

**A. DATASETS**

1) OUR DATASET

Our dataset contains 7760 aerial images and 4355 of them are used as training set while the rest forms testing set. There are totally 312071 vehicles and 171556 of them are included in training set and the rest vehicles are included in testing set. In COCO dataset [14], small object refers to the object whose size is smaller than $32 \times 32$, The size of a medium object is between $32 \times 32$ and $96 \times 96$, while the size of a large object is larger than $96 \times 96$. A typical resolution of an image in COCO dataset is $640 \times 427$, whereas the resolutions of most UAV images are $3840 \times 2160$. Therefore, it is not suitable to use the 'size' definition of objects with the standard of COCO dataset. The scale distribution of vehicles in our dataset is illustrated in Figure 7. The definition of area ratio is shown in Eq.(9). We utilize the area ratio of target to classify the objects into three categories: small objects, medium objects and large objects. We consider vehicles with area ratio in $(0, 0.06\%]$ as small vehicles. Vehicles with area ratio in $(0.06\%, 0.1\%]$ as medium vehicles while the area ratio of large vehicles is in $(0.1\%, 1]$.

2) UAVDT DATASET

The UAVDT [46] dataset contains 40409 images and of training data and 23829 of them are used as training set while the rest forms testing set. The resolutions of images are about $1080 \times 540$ pixels. The categories in UAVDT are car, bus and truck. The quantity distribution of different categories is shown in Figure 8.

**B. EVALUATION METRICS**

1) OUR DATASET

Popular metrics including: precision [41], recall rate [41], F1-Score [47] and AP (average precision) are used for...
performance evaluation. AP refers to the area under the precision-recall curve. The definition of other metrics is shown in Eq.(14)–(16).

\[
Recall = \frac{TP}{TP + FN} \quad (14)
\]
\[
Precision = \frac{TP}{TP + FP} \quad (15)
\]
\[
F1 – Score = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (16)
\]

where \( TP \) denotes true positive which defines the total number of positive detections that are really positive, \( FN \) represents false negative which defines the number of vehicles that are not detected, \( FP \) indicates false positive which defines the number of false alarms.

2) UAVDT DATASET
Mean average precision (mAP) is utilized to evaluate the performance of each algorithm on UAVDT. It is the average APs of different categories. The mAP is high related to IoU. Therefore, \( mAP_{0.5} \), \( mAP_{50} \) and \( mAP_{75} \) is utilized the metrics. Specifically, \( mAP_{50} \) and \( mAP_{75} \) refer to mAP at \( IoU = 0.5 \) and \( IoU = 0.75 \) respectively while \( mAP_{0.5} \) refers to the average value when \( IoU = [0.5 : 0.05 : 0.95] \).

C. SENSITIVITY ANALYSIS
OA-NMS has three hyper-parameters including \( \sigma \), \( \alpha \), \( \beta \) and a score threshold \( N_t \). We vary these parameters and measure recall, precision and F1-Score on the training set. The IOU threshold between candidate detections and ground truth is set to 0.5, which means that only if the overlap is higher than 0.5, the candidate detection is a true positive detection. First of all, we set \( \xi \) to 0.1, such that it can reduce the distortion of objects and repeated detections caused by image cropping.

|       | NMS | SoftNMS | OA-NMS |
|-------|-----|---------|--------|
| \( N_T \) | 0.2 | —       | —      |
| \( N_t \) | —   | —       | 0.45   |
| \( \sigma \) | —   | 0.05    | 0.1    |
| \( \alpha \) | —   | —       | 0.95   |
| \( \beta \) | —   | —       | 0.3    |
FIGURE 11. Detection results of the original SSD and SSP-SSD. The green boxes are true positives. The yellow boxes are false positives. The red boxes are missed detections. First row: the detection results of the original SSD. Second row: results of SSP-SSD.

To find a fine parameter set, we vary these parameters from 0.05 to 0.95 and apply grid search to obtain the values as listed in Table 1.

To demonstrate the relation among these parameters and the metrics, we first set $\alpha$ and $\beta$ to values given in Table 1. Under these circumstances, we obtain the influence of $\sigma$ and $N_t$ on the model performance, as demonstrated in Figure 9. It can be seen that the smaller $N_t$ is, the higher the recall is. This means that we can obtain more true detections at a lower value of $N_t$. Meanwhile, the smaller $N_t$ is, the smaller the precision is. It indicates that when the value of $N_t$ increases, the false detections/alarms tend to decrease. According to Eq.(8) and Figure 9, when $\sigma$ increases, the OA-NMS scores tend to be less sensitive to IoU scores which result in a higher recall rate and smaller precision rate. In addition, we assign the values of $N_t$ and $\sigma$ according to Table 1 and vary $\alpha$ and $\beta$ from 0.05 to 0.95 to obtain Figure 10. It can be seen from Figure 6 and 10 that the relationship between the performance and the two parameters $\alpha$ and $\beta$ is similar to that with $\sigma$.

SoftNMS has a hyper-parameter $\sigma$ and its sensitivity on recall, precision and F1-score is shown in Figure 12 (a). It can be seen that the recall is stable when $\sigma$ increases while precision drops significantly in [0.3, 0.45]. It is mainly because that SoftNMS is less sensitive to IoU when $\sigma$ increases and more redundancies are retained. NMS sensitivity against the variation of overlap threshold $N_T$ is shown in Figure 12 (b). As $N_T$ increases, recall also increases while precision decreases.

To achieve a trade-off between recall and precision, we set parameters shown in Table 1.

D. ABDATION STUDY

1) SSP EVALUATION

We propose SSP to construct effective representations for multi-scale vehicles. In Figure 11 and Table 2, we compare some detection results of SSP-SSD and those of the original
TABLE 3. Performance comparison of different result fusion algorithms.

| Algorithm                  | Backbone | Recall | Precision | F1-Score | Average Speed (ms) |
|----------------------------|----------|--------|-----------|----------|-------------------|
| SSP-SSD+NMS [39]           | Resnet101| 0.8335 | 0.9385    | 0.8829   | 76                |
| SSP-SSD+SoftNMS [45]       | Resnet101| 0.8082 | 0.9680    | 0.8809   | 77                |
| SSP-SSD+OA-NMS             | Resnet101| 0.8188 | 0.9721    | 0.8889   | 79                |
| IP+SSP-SSD+NMS [39]        | Resnet101| 0.9168 | 0.8568    | 0.8858   | 95                |
| IP+SSP-SSD+SoftNMS [45]    | Resnet101| 0.9212 | 0.8520    | 0.8852   | 97                |
| IP+SSP-SSD+OA-NMS          | Resnet101| 0.8855 | 0.9475    | 0.9154   | 101               |

FIGURE 13. Demonstration of fusion results of three methods. The green boxes are true positives. The yellow boxes are false positives. The red boxes are missed detections. First row: the detections of the original images and its patches. Second row: results of NMS. Third row: results of SoftNMS. Forth row: results of OA-NMS.

SSD. It can be seen that the SSP significantly improves the detection performance for small vehicles. Meanwhile, thanks to SSP, false alarms also decrease, which means the context information is able to enhance the feature representation of objects. The detection performance on large vehicles has been improved by SSP, but it is not as obvious as small vehicles. The main reason is that the detector utilizes high-level features to represent large objects which describe the object better than low-level features and weaken the contribution of context information.
2) EVALUATION OF DIFFERENT FUSION ALGORITHMS

We report the performance of different methods in Table 3 when overlap threshold is 0.5. It can be seen that the image preprocessing has a significant improvement on the recall value which means it is effective in dealing with IoU distribution imbalance problem. However, it also introduces many false detections. During the result fusion stage, SoftNMS achieves the highest recall with lowest precision among the three algorithms, which is mainly because it aims to keep more detections with high overlap while the intersection between the bounding boxes of most vehicles is relatively small in aerial images. In contrast, OA-NMS obtains the highest precision due to the removal of outliers. In terms of F1-Score, OA-NMS achieves the best performance.

E. PERFORMANCE EVALUATION

1) EXPERIMENTS ON OUR DATASET

In this section, we compare our algorithm with other SOTA methods including: SSD [23], Cascade RCNN [19], FRCNN [17], YOLOV3 [22], YOLOV4 [51], YOLOV5(x) [52], FCOS [49], Retinanet [50] and CenterNet [48]. Resnet101 is selected as backbone for most of these networks except that YOLOV3 adopts Darknet53 [22] as backbone while YOLOV4 and YOLOV5 adopt CSPDarknet [51] as...
backbone. We analyze the performance of each model using different overlap thresholds. The results demonstrated in Figure 14 indicate that our method obtains the highest recall, precision and F1-Score. In Table 4, we list the performance of each method when overlap threshold equals 0.5. It can be seen that our method achieves the best performance in terms of AP and F1-Score. As for computation efficiency, it can be seen from Table 3 that, it takes SSP-SSD 79ms to finish detecting the original image. Therefore, it will take SSP-SSD much more time to finish detecting the original image and two cropped patches serially. However, owing to parallel computation, it only takes our method 101ms to deal with one image which saves much time compared with serial computation.

In addition, we report the detection performance of each model on small, medium and large vehicles respectively, when overlap threshold is 0.5 (as shown in Table 5). For medium and large vehicles, these algorithms can achieve SOTA performance except CenterNet which mainly relies on single scale heatmap. However, for small vehicles, low-level features extracted by SSD lack the context information so they can not effectively represent small objects. Although YOLOv3 introduced high-level semantic information for feature representation of small vehicles, it may include too much background information. Owing to the cascade structure, the Cascade RCNN is able to learn high quality features for multi-scale vehicles which makes it perform better than FRCNN. It can be seen that our method can achieve SOTA performance on multi-scale vehicles.

Finally, we report the performance of other methods when utilizing OA-NMS to remove redundant detections (as shown in Table 4). It can be seen that OA-NMS can improve the precision of these methods while it will lead to decreasing in recall rate. It mainly because OA-NMS is a re-score method which is sensitive to the outliers with extreme scales or aspect ratios. Therefore, OA-NMS may remove some true detections with low scores when removing false detections. If there are a large number of outliers, the method will greatly improve the performance of the method in terms of F1-Score and AP, such as CenterNet and YOLOv4. However, if the number of outliers is small, it would not obtain an obvious improvement in F1-Score and AP such as Faster RCNN, Cascade RCNN and SSD. Even worse, the F1-Score and AP may decrease a little such as FCOS, Retinanet, YOLOv3 and YOLOv5.

2) EXPERIMENTS ON UAVDT DATASET

We compared our method with other SOTA methods on UAVDT dataset including: R-FCN [18], RON [53], SSD [23], Faster RCNN (FRCNN) [17], ClusDet [54] and DMNet [38]. All experiment results of other SOTA methods are obtained.
from [38] and the detection performance is illustrated in Table 6. It can be seen that our method outperforms the SOTA methods except DMNet. However, it will cost DMNet 290ms to process one image which is much slower than the proposed method. In addition, it can be seen that the performance of these detectors on UAVDT [46] is much worse than that on our dataset. The main reason of that is the unbalanced data.

V. CONCLUSION

Vehicle detection in aerial images remains a challenging problem due to IoU distribution imbalance problem and the difficulty of learning feature representation of vehicles under different scales. Therefore, in this paper, we propose a robust vehicle detection model for aerial images. First, we perform image preprocessing to deal with IoU distribution imbalance problem and greatly improve the recall rate. Then, we propose SSP-SSD to enhance feature representation of vehicles with different scales and improve the precision. From the experiments, it can be seen that the SSP-SSD is able to learn effective features for multi-scale vehicles. To reduce the detection time, we combine the original image and its patches into a batch and utilize parallel computation to accelerate the detection process. However, it still cost much time compared with detecting only on the original image. The main reason is that the computation burden increases when dealing with three images. Therefore, a more efficient backbone can be designed to improve the time efficiency. Finally, we propose OA-NMS to fuse results and remove false detections with extreme scales or aspect ratios. Although OA-NMS can remove many false detections, it can also remove some true positives if they have low scores and extreme scales or aspect ratios. In the future, we will focus our research on few-shot learning to improve the performance of our detector on unbalanced data.

ACKNOWLEDGMENT

The authors are very grateful for all anonymous reviewers who give us many valuable comments and suggestions and help us improve and present this paper.

REFERENCES

[1] K. Uto, H. Seki, G. Saito, and Y. Kosugi, “Characterization of rice paddies by a UAV-mounted miniature hyperspectral sensor system,” IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens., vol. 6, no. 2, pp. 851–860, Apr. 2013.
[2] V. Sagan, M. Maimaitijiang, P. Sidike, K. Eblimit, K. Peterson, S. Hartling, F. Esposito, K. Khanal, D. Pauli, R. Ward, F. Fritschi, N. Shakoor, and T. Mockler, “UAV-based high resolution thermal imaging for vegetation monitoring, and plant phenotyping using ICI 8640 P, FLIR vue pro r 640, and thermoMap cameras,” Remote Sens., vol. 11, no. 3, p. 330, Feb. 2019.
[3] H. Long, Y. Chung, Z. Liu, and S. Bu, “Object detection in aerial images using feature fusion deep networks,” IEEE Access, vol. 7, pp. 30980–30990, Apr. 2019.
[4] B. Ma, Z. Liu, F. Jiang, Y. Yan, J. Yuan, and S. Bu, “Vehicle detection in aerial images using rotation-invariant cascaded forest,” IEEE Access, vol. 7, pp. 59613–59623, 2019.
[5] R. Ke, Z. Li, J. Tang, Z. Pan, and Y. Wang, “Real-time traffic flow parameter estimation from UAV video based on ensemble classifier and optical flow,” IEEE Trans. Intell. Transport. Syst., vol. 20, no. 1, pp. 54–64, Jan. 2019.
[6] M. Bejiga, A. Zeggada, A. Nouffidj, and F. Melgani, “A convolutional neural network approach for assisting avalanche search and rescue operations with UAV imagery,” Remote Sens., vol. 9, no. 2, p. 100, Jan. 2017. [Online]. Available: http://www.mdpi.com/2072-4292/9/2/100
[7] A. Kembhavi, D. Harwood, and L. S. Davis, “Vehicle detection using partial least squares,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 6, pp. 1250–1265, Jun. 2011.
Z. Cai, C. Wang, H. Luo, H. Wang, Y. Chen, C. Wen, Y. Yu, L. Cao, and J. Li, “Vehicle detection in high-resolution aerial images based on fast sparse representation classification and multiorder feature,” IEEE Trans. Intell. Transport. Syst., vol. 17, no. 8, pp. 2296–2309, Aug. 2016.

N. Amour, H. Alhichri, Y. Bazi, B. Benjdira, N. Alajlan, and M. Zaat, “Deep learning approach for car detection in UAV imagery,” Rem. Sens., vol. 12, no. 4, p. 312, Mar. 2017. [Online]. Available: http://www.mdpi.com/2072-4299/12/4/312

M. ElMikaty and T. Stathaki, “Car detection in aerial images of dense urban areas,” IEEE Trans. Aerosp. Electron. Syst., vol. 54, no. 1, pp. 51–63, Feb. 2018.

W. Li, H. Li, Q. Wu, X. Chen, and K. N. Ngan, “Simultaneously detecting and counting dense vehicles from drone images,” IEEE Trans. Ind. Elec., vol. 66, no. 10, pp. 7651–7662, Dec. 2019.

M. ElMikaty and T. Stathaki, “Detection of cars in high-resolution aerial images of complex urban environments,” IEEE Trans. Geosci. Remote Sens., vol. 55, no. 10, pp. 5913–5924, Oct. 2017.

T. Moranduzzo and F. Melgani, “Categorizing cars in UAV images with a catalog-based approach,” IEEE Trans. Geosci. Remote Sens., vol. 52, no. 10, pp. 6536–6567, Oct. 2014.

T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common objects in context,” CoRR, vol. abs/1405.0312, pp. 740–755, Sep. 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

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.

R. Girshick, “Fast R-CNN,” in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Dec. 2015, pp. 1440–1448.

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.

J. Dai, Y. Li, K. He, and J. Sun, “R-FCN: Object detection via region-based fully convolutional networks,” in Proc. 30th Conf. Adv. Neural Inf. Process. Syst. (NIPS), Barcelona, Spain, vol. 29, D. D. Lee, and M. Sugiyama, and U. V. Luxburg, and I. Guyon, and R. Garnett, Eds. Los Angeles, CA, USA: Curran, 2016.

Z. Cai and N. Vasconcelos, “Cascade R-CNN: Delving into high quality object detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 6154–6162.

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.

J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 6517–6525.

J. Redmon and A. Farhadi, “Yolov3: An incremental improvement,” CoRR, vol. abs/1804.02767, pp. 1–6, Apr. 2018. [Online]. Available: http://arxiv.org/abs/1804.02767

W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” vol. abs/1512.02325, pp. 1–6, Apr. 2018. [Online]. Available: http://arxiv.org/abs/1512.02325

K. Oksuz, B. C. Cam, S. Kalkan, and E. Akbas, “Imbalance problems in object detection: A review,” IEEE Trans. Pattern Anal. Mach. Intell., early access, Mar. 19, 2020, doi: 10.1109/TPAMI.2020.2981890.

F. O. Unel, B. O. Ozkalayci, and C. Cigla, “The power of tiling for small object detection,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2019, pp. 582–591.

X. Cao, C. Wu, P. Yan, and X. Li, “Linear SVM classification using boosting HOG features for vehicle detection in low-altitude airborne videos,” in Proc. 10th IEEE Int. Conf. Image Process. Ser., Sep. 2011, pp. 2421–2424.

T. Moranduzzo and F. Melgani,“A SIFT-SVM method for detecting cars in UAV images,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Jul. 2012, pp. 6868–6871.

H. Zhou, L. Wei, C. P. Lim, D. Creighton, and S. Nahavandi, “Robust vehicle detection in aerial images using Bag-of-Words and orientation aware scanning,” IEEE Trans. Geosci. Remote Sens., vol. 56, no. 12, pp. 7074–7085, Dec. 2018.

T. Moranduzzo and F. Melgani, “Comparison of different feature detectors and descriptors for car classification in UAV images,” in Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS), Jul. 2013, pp. 204–207.

W. Shao, Y. Yang, G. Liu, and J. Liu, “Car detection from high-resolution aerial imagery using multiple features,” in Proc. IEEE Int. Geosci. Remote Sens. Symp., Jul. 2012, pp. 4379–4382.

N. Audebert, B. Le Saux, and S. Lefèvre, “Segment-before-detect: Vehicle detection and classification through semantic segmentation of aerial images,” IEEE Trans. Geosci. Remote Sens., vol. 57, no. 10, pp. 7339–7351, Oct. 2019.

T. Sommer, T. Schaulin, and J. Beyerer, “Comprehensive analysis of deep learning-based vehicle detection in aerial images,” IEEE Trans. Circuits Syst. Video Technol., vol. 29, no. 9, pp. 2733–2747, Sep. 2019.

Y. Koga, H. Miyazaki, and R. Shibasaki, “A CNN-based method of vehicle detection from aerial images using hard example mining,” Remote Sens., vol. 10, no. 1, p. 124, 2018. [Online]. Available: https://www.mdpi.com/2072-4299/10/1/124

T. Moranduzzo and F. Melgani, “Scale-aware trident networks for object detection,” in Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV), Oct. 2019, pp. 6603–6606.

A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” 2020, arXiv:2004.10934. [Online]. Available: http://arxiv.org/abs/2004.10934
[52] G. Jocher, A. Stoken, J. Borovec, NanoCode012, ChristopherSTAN, L. Changyu, Laughing, A. Hogan, lorenzomammana, tkianai, yxNONG, AlexWang1900, L. Diaconu, Marc, wanghaoyang0106, mlsah, Doug, Hatovix, J. Poznanski, L. Yu, changyu98, P. Rai, R. Ferriiday, T. Sullivan, W. Xinyu, YuriRibeiro, E. R. Claramunt, hopesala, pritul dave, and yzchen. (Aug. 2020). Ultralytics/YOLOV5: V3.0. [Online]. Available: https://github.com/ultralytics/yolov5, doi: 10.5281/zenodo.3983579.

[53] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, “RON: Reverse Connection with Objectness Prior Networks for Object Detection,” in Proc. 30TH IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Honolulu, HI, USA, New York, NY, USA: IEEE, Jul. 2017, pp. 5244–5252.

[54] F. Yang, H. Fan, P. Chu, E. Blasch, and H. Ling, “Clustered object detection in aerial images,” CoRR, vol. abs/1904.08008, pp. 8311–8320, Oct. 2019. [Online]. Available: http://arxiv.org/abs/1904.08008

[55] 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. 936–944.

XINDE LI (Senior Member, IEEE) received the Ph.D. degree in control theory and control engineering from the Department of Control Science and Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2007. Afterwards, he joined the School of Automation, Southeast University, Nanjing, China, where he is currently a Professor and a Ph.D. Supervisor. From 2012 to 2013, he was a Visiting Scholar with the School of Interactive Computing, Georgia Institute of Technology. In 2016, he was a Postdoc Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore. His research interests include information fusion, object recognition, computer vision, intelligent robot, and human–robot interaction. He was granted a “Talent of Qing Lan Project” award of Jiangsu province and a “Six Major Top-Talent Plan” award of Jiangsu province, China.

HONG PAN (Member, IEEE) received the Ph.D. degree in pattern recognition and intelligent systems from Southeast University in 2004. He is currently an Associate Researcher with the School of Automation, Southeast University. From September 2004 to August 2006, he worked as a Research Associate with the Multimedia Signal Processing Center, Hong Kong Polytechnic University, where he worked on image sparse transform and coding. His research interests include machine learning, deep learning, computer vision, medical image processing and analysis, and multimedia signal processing (image/video codec, retrieval and analysis).

VOLUME 8, 2020

X. Li et al.: Multi-Scale Vehicle Detection in High-Resolution Aerial Images With Context Information

XIANGHUI LI was born in Yancheng, Jiangsu, China, in 1992. He received the B.Sc. degree in automation from Southeast University, Nanjing, Jiangsu, China, in 2014, where he is currently pursuing the Ph.D. degree in pattern recognition. His research interests include deep learning and place identification.

XINDE LI (Senior Member, IEEE) received the Ph.D. degree in control theory and control engineering from the Department of Control Science and Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China, in 2007. Afterwards, he joined the School of Automation, Southeast University, Nanjing, China, where he is currently a Professor and a Ph.D. Supervisor. From 2012 to 2013, he was a Visiting Scholar with the School of Interactive Computing, Georgia Institute of Technology. In 2016, he was a Postdoc Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore. His research interests include information fusion, object recognition, computer vision, intelligent robot, and human–robot interaction. He was granted a “Talent of Qing Lan Project” award of Jiangsu province and a “Six Major Top-Talent Plan” award of Jiangsu province, China.

HONG PAN (Member, IEEE) received the Ph.D. degree in pattern recognition and intelligent systems from Southeast University in 2004. He is currently an Associate Researcher with the School of Automation, Southeast University. From September 2004 to August 2006, he worked as a Research Associate with the Multimedia Signal Processing Center, Hong Kong Polytechnic University, where he worked on image sparse transform and coding. His research interests include machine learning, deep learning, computer vision, medical image processing and analysis, and multimedia signal processing (image/video codec, retrieval and analysis).

VOLUME 8, 2020

X. Li et al.: Multi-Scale Vehicle Detection in High-Resolution Aerial Images With Context Information

XIANGHUI LI was born in Yancheng, Jiangsu, China, in 1992. He received the B.Sc. degree in automation from Southeast University, Nanjing, Jiangsu, China, in 2014, where he is currently pursuing the Ph.D. degree in pattern recognition. His research interests include deep learning and place identification.