2022 14th International Conference on Computational Intelligence and Communication Networks (CICN) | 978-1-6654-8771-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/CICN56167.2022.10008383

Small Object Difficulty (SOD) Modeling for Objects Detection in Satellite Images

Debojyoti Biswas
Computer Science
Texas State University
San Marcos, TX US

Jelena Tešić
Computer Science
Texas State University
San Marcos, TX US

***Abstract*—Recent increases in aerial image access and volume, increases in computational power, and interest in applications have opened the door to scaling up object detection to production. Aerial data sets are very large in size, and each frame of the data set contains a huge number of dense and small objects. Deep learning applications for aerial imagery are behind due to a high variety between datasets (e.g. object sizes, class distributions, object feature uniformity, image acquisition, distance, weather conditions), the size of objects in satellite imagery, and the subsequent failure of state-of-the-art architecture to capture small objects, local features, and region proposals for densely overlapped objects in satellite images. In this paper, we provide a novel pipeline that improves the backend through spatial pyramid pooling, a partial cross-stage network, a region proposal network via heatmap-based region proposals, and object localization and identification through a novel image difficulty score that adapts the overall focal loss measure based on the image difficulty. Our proposed model outperformed the state-of-the-art method in mAP by 1.8% and 2.3% in the DOTA and DIOR data sets, respectively.**

I. Introduction

The next frontier in precision agriculture, emergency rescue system, terrestrial and naval traffic monitoring, and industrial surveillance is the integration of automated, reliable object location from overhead satellite and aerial imagery [1], [2]. Continuous improvements in deep neural network (DNN) models, combined with increased access to computational resources, have enabled the improvement of object detection methods in both aerial [3]–[5] and consumer images [6]–[9]. For DNNs to make a reliable localization in satellite imagery, the networks require large and diverse amounts of training data. There are only a handful of reliably annotated datasets for overhead imagery, e.g: DOTA2.0 [4], DIOR [5], Visdrone [10], and very few reliable object detection methods in the satellite imagery approach [4], [11], [12]. The shortcoming of these works on satellite images are as follows: considering complex datasets with very small and dense objects, considering complex background, large number of objects per image, per image class variation, handling hard examples, and performing better local feature extraction for small objects.

Typical object detection approaches developed for consumer images fail on the satellite imagery due to the relative object size to image size and the number of objects in the image. The underpinning assumption in the state-of-the-art (SOTA) architecture design for object detection is that the number of objects in an image is in the single digits and the object size is greater than 1% of the image size. Figure 1 illustrates typical examples for consumer and for satellite images.

Fig. 1. Consumer and aerial image examples

Satellite images are taken from high altitudes. The number of objects per image is usually in the triple digits, and the size of the object is often less than 0.01% of the size of an image. The size of the satellite image is up to 400 million pixels, and the sizes of objects are often less than 100 pixels. A typical satellite image patch is 1024×1024 or 1.05 million pixels. If an object is 10×10 or 100 pixels, the size of the object is < 0.0001 of the area of the image. Small objects in satellite images tend to be densely packed, and the success of object detection depends on how reliable the pixel- and object- feature extraction and region proposal network in the DNN architecture is. The increased number of very small densely packed objects in the image increases the chance of losing pixel-level feature information during the feature extraction phase. The RPN-based proposal network misses a large number of small objects in the early stage of the processing pipeline and cannot be recovered in the detection stage [14]. Comparison between RPN, One-Stage and Heatmap based object detectors and the heatmap based network [15] often outperforms the RPN-based network [9], [16] in small object detection [14].

In this paper, we have developed an object detection network tuned to satellite imagery. Section II summarizes related work. Section III introduces the proposed methodology and improvements to the object detection training pipeline where we design a strong darknet-style [17] backbone based on spatial pyramid pooling (SPP) and the partial cross-stage network (CSP) [18], followed by a heatmap-based region proposal generator (RPG) to address the challenge of small dense objects in satellite imagery. The heatmap-based proposal box regression completely

978-1-6654-8771-9/22/$31.00 ©2022 IEEE
DOI: 10.1109/cicn.2022.21

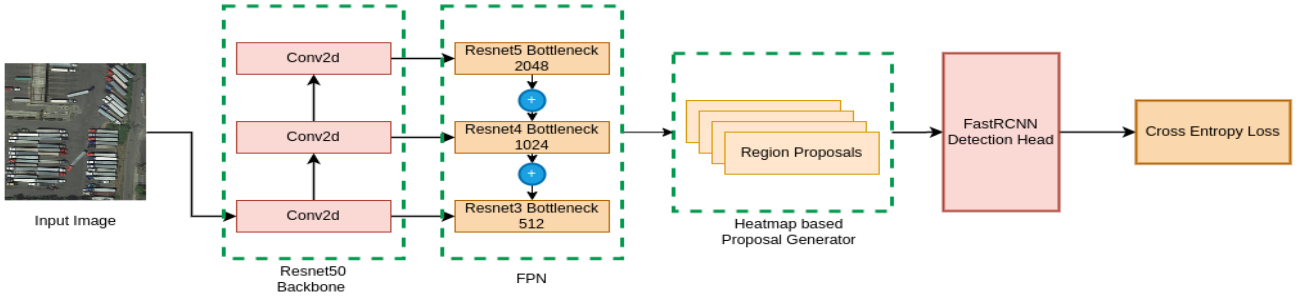


Fig. 2. *Base* model: heatmap based multi-stage object detection model [13].

eliminates the costly nonmaximal suppression step [13] in the RPN post-processing. In Section IV we present the experimental results on satellite image sets and show that the proposed approach outperforms the available state-of-the-art (SOTA) models in terms of detecting overlapped and small objects, and producing higher mAP across all classes. We present future work in Section V.

II. Related Work

State-of-the-art (SOTA) object detectors are single-stage or multistage detectors. Single-stage detectors are object detection networks without an RPN module [8], [17]. They are mainly based on different scale and aspect ratios of the anchor boxes. Single-stage object detection architectures were shown to miss a significant amount of small objects in satellite imagery, and they also require a good anchor design for better performance [4], [19]. Although single-stage detectors were able to achieve state-of-the-art results for consumer images, the region proposals from single-stage detectors are over-dominated by the negative examples, which makes the detection layers biased toward a False Positive result during the training. Multistage detectors are often more reliable [20], [21] because of extra effort to improve the image regions of interest. Multistage detectors [9], [16] use RPN to filter out positive instances from the image with the help of IOU and the non-maximum suppression technique (NMS). With multistage detectors, we have to train an additional network for region generations, though there have been several works for a lighter version of object detection which does not require the RPN network [6], [8], [17].

Heatmap-based RPN uses a Gaussian filter which creates a heatmap peak at the center of the object to define proposal regions. The center is used here as an anchor to the object and is based solely on location, not box overlap [15]. Therefore, we have only one anchor per object, which eliminates the heavy usage of NMS to filter overlapped proposals, without affecting the quality of the proposal. As the size of the object decreases, the chances of losing local information in deep layers increase significantly. The first step introduced toward small detection was the use of FPN [22]. Instead of relying on a single scale feature, it was proposed to use different scale features from different stages of the backbone for different scale predictions. However, the performance of the FPN network still depends on a strong backbone network. The backbone network produces bottom-up features from the input image, and the FPN layer upscales the bottom-up features with the combination of lateral connections to create top-bottom features for scale prediction. FPN also helps to strengthen low spatially rich features by combining semantically rich features. Subsequently, several other variants [23], [24] of FPN method were formed for improved performance. The pixel-level appearance features do not contain enough information to localize small objects in an image, and recent research shows that the context-based bidirectional feature fusion of neighboring pixels helps to localize small objects in an image [25].

III. Methodology

A. Base Model

The base model used in this paper was recently introduced in [13], and is illustrated in Figure 2. We have used this model as our reference model and adjusted *image load size*, *number of output channels per CNN block* and *IOU in FastRCNN Detection Head* parameters for the satellite data set. The base model has three different parts: Backbone, Region Proposal Network (RPN), and Detection Head.

Backbone Backbone combines ResNet50 [13] as a feature extractor and the feature pyramid network (FPN) for multi-scale prediction. The residual connection efficiently combines features from previous layers with skip connections [26]. The residual block architecture allows for a deeper model without the vanishing-gradient effect. ResNet50 [13] achieves state-of-the-art performance in the COCO [27] and LVIS [21] data sets. The FPN layer extracts three different scales of features from different layers of the backbone network, and the strides applied on these three scales are 8, 16, and 32, respectively, as illustrated in Figure 2. Also in Figure 2, the Resnet3, Resnet4 and Resnet5 blocks represent strides of 8, 16, and 32 respectively.

Region Proposal Network (RPN) RPN integrates the heatmap-based region proposal network [15] into the *Base* model object detection pipeline. ResNet Style RPN degrades performance in satellite imagery because there is a large overlap between objects, and there is a significant number of small objects per image [8]. Also, the number of non-maximal suppression computations in every pair of proposals per image is 1000+ times higher for satellite imagery than for consumer imagery. The most efficient way is to use probabilistic region proposals. Probabilistic

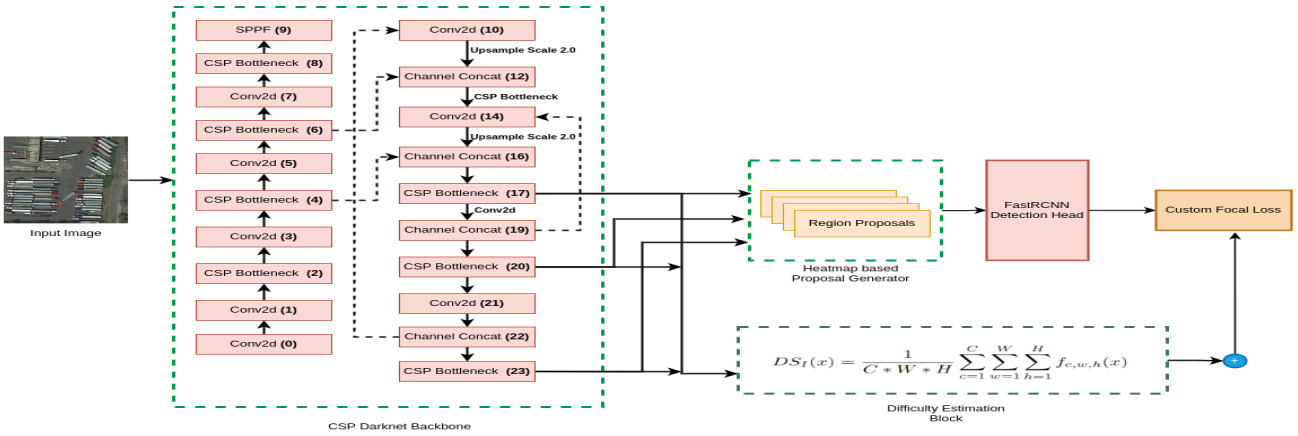


Fig. 3. SOD Model Architecture: Small Objectness & Diifficulty improvements to the DNN object detection pipeline: SOD architecture

region proposals are calculated using Gaussian kernels on the different scales of features outputted by the backbone element [15]. The element-wise comparison between the max-pool input and the output of the Gaussian kernel produces a heatmap. The max-pool operation will elevate each pixel in the feature, except for the local maxima, where the value is 1. Each peak in the heatmap corresponds to a keypoint of the object (center). Then, the image features at each keypoint are used to predict the height and width of each object, and the resulting regressed bounding boxes are shown to perform well when objects are close to each other and overlap. As a computational bonus, the proposed RPN enables the detection forward-pass, and we can skip the non-maximal suppression step in our pipeline. We have adjusted *image augmentation size*, *number of proposals per image*, and *heatmap minimum overlap per object* in this block to satellite images.

Detection Head Detection head is adapted from the Faster-RCNN detector [9]. The detection head takes as input the filtered region proposals from the RPN module. The RPN provides different scale proposals at different steps, as discussed in the backbone module. The first task in the detection head is to convert each proposal into 7×7 pixel size grids with the same number of channels using the region-of-interest (ROI) pooler. Then, the ROI pooler output is flattened and fed into the fully connected network (FCN) layers. The final stage outputs a (N, C) class predictor for C classes and N region proposals, and $(N, 4)$ bounding boxes [9]. We have also adjusted the parameters *number of detection per image* and *IOU at detection block* for satellite imagery.

B. SOD Model

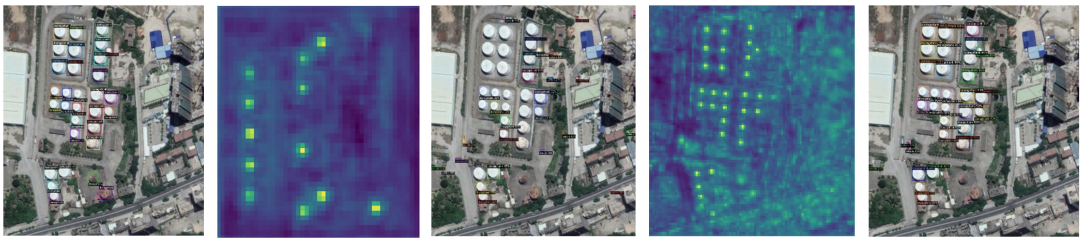
We propose the new SOD Model, short for Small Objectness and Difficulty (SOD) Modeling Improvements, illustrated in Figure 3. First, we add the CSP Darknet backbone, then add the *Difficulty Estimation* block, and finally change the cross-entropy loss with a modified version of focal loss, as illustrated in Figure 3.

Backbone The efficiency of the *RPN* module depends on the effectiveness of the *Backbone* module. If the backbone fails to extract meaningful features for the small object in the image, the RPN module will likely fail to include the small object in the region proposals. In *Base Model*, ResNet50 and ResNet101 feature extractors have not been able to extract meaningful features from satellite images that contain many small objects, as illustrated in Figure 4(b) and Figure 7(b). The deeper layers of CNN architectures use a larger number of steps. This approach increases semantic information but loses spatial information in the feature extraction module. On the other hand, the darknet backbone uses the partial split network of stages to preserve better semantic information in the deeper layers of CNN [17]. First, we propose to integrate the partial cross stage (*CSP darknet*) [18] as a new backbone since it offers aggregation layers at low and high resolution. In the next step, we propose replacing the max-pooling layer with the spatial pyramid-pooling layer for finer feature extraction. Proposed SOD model for the detection of small objects in satellite imagery is illustrated in Figure 3: the concatenation of layers between layers 6 and 12, layer 4 and 16, layer 14 and 19 and layer 10 and 22 propagates the information from the lower level to the higher level. The RPN module takes features from Layers 17, 20, and 23 for proposal generation, as illustrated in Figure 3. Since the objective is to detect and identify small and dense objects in satellite imagery, we introduce two new blocks into the pipeline: *Custom Focal Loss* and *Difficulty Estimator Block*, as illustrated in Figure 3.

Difficulty Estimator (DE) Here, the DE module derives an image feature's complexity from the network's active neuron information. The difficulty score (DS) for a FPN feature level with a resolution of $C \times W \times H$ for the image I is calculated in Eq. 1.

$$DS(I) = \frac{1}{C*W*H}\sum_{c=1}^{C}\sum_{w=1}^{W}\sum_{h=1}^{H} f_{c,w,h}(I) \quad (1)$$

Here, *C, W, H* are feature output channels, feature width, and feature height, respectively, at any FPN level; $f_{c,w,h}(I)$ denotes the value from Sigmoid Linear Unit (SiLU) at every pixels in the image *I*. Using this block, we calculate the number of total neurons fired for a single image in the



(a) Ground Truth (b) Heatmap Base (c) Base Prediction (d) Heatmap SOD (e) SOD Prediction

Fig. 4. heatmap-based region proposal pipeline results for DIOR data

forward pass. We sum up all activation values and divide them by the total dimension of the characteristic *C, W, H* to obtain the difficulty score (DS) at the FPN level. We derive this DS from 3 different FPN levels and average the values to obtain the final DS for an image *I*. The increase in complexity for this DS block is nearly negligible. The Big Oh *(O)* notation for this block is *O(r)*, where r is the batch size in each iteration.

$$\forall c \in C, \alpha'_c = -1 * log\left(\frac{|C_c|}{|C_1 \cup C_2 \cup ...|}\right) \\ \Rightarrow \alpha_c = \beta * \frac{\alpha'_c - min(\alpha_c)}{max(\alpha_c) - min(\alpha_c)} \tag{2}$$

a) Custom Focal Loss: The increased number of trivial examples in the region proposals incurs a small amount of loss for every example, which in turn contributes significantly while using cross-entropy as a loss function. Custom focal loss is calculated from the difficulty scores for each image, and we propose replacing the loss of cross-entropy with the loss of custom focalization, as illustrated in Figure 3. This adjustment reduces the impact of the class labeling imbalance and the trivial/difficult object identification span for objects in the satellite imagery data set. In the proposed focal loss function, we use difficulty scores calculated for each image by a difficulty estimator block as a weight factor to focus more on complex images with a high diversity of objects and a high variation in pixel-level features. The basic form of the focal loss function is $FL(p_t, y) = \alpha_t * (1 - p_t)^\gamma * CE(p, y)$, where p_t is the probability distribution of the target t, and y is the ground truth of the object being a specific class, γ is the modulating factor, α_t is used as a weighting factor and CE represents cross-entropy function. We propose a new measure, Difficulty Weighted Focal Loss (DWFL) and define it as a product of difficulty score and focal loss for the image $DWFL(x, p, y) = DS(I) * FL(p, y)$, where $DS(I)$ is defined in Eq. 1. The value α is used in the DWFL calculation to control the class imbalance problem in our source and target data sets. Here, the value of the parameter α increases if the frequency of a particular class is very low and decreases if the frequency of a particular class is very high. In this way, we focus more on minor classes. The α_c is calculated as in Eq. 2 for each class, where the modulating factor α'_c depends on the frequency $|C_c|$ of a particular class in the data set and $|C_1 \cup C_2 \cup C_3 ...|$ is the total number of all instances of all classes in the data set.

We use these normalized α_c values from Eq. 2 across different classes c, $c \in C$ to mitigate the imbalance of class labeling. The scaling factor $\beta = 0.6$ was found to be the most appropriate for satellite imagery. The range of α'_c values in the DIOR data set is 0.2 to 0.79 and the range of α'_c values in the DOTA data set is 0.15 to 0.96, which represents a very tight scaling factor for *FL* in both data sets. The proposed normalization of α_c in Eq. 2 is more effective and gives a stable loss calculation for a highly unbalanced class count in the data set.

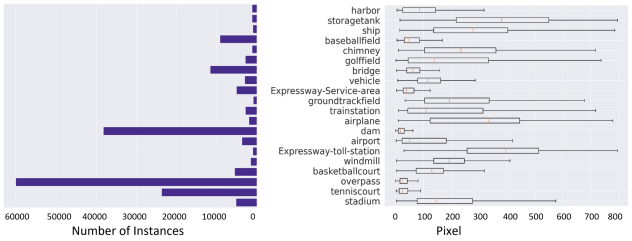


Fig. 5. DIOR [5] class and object size distribution.

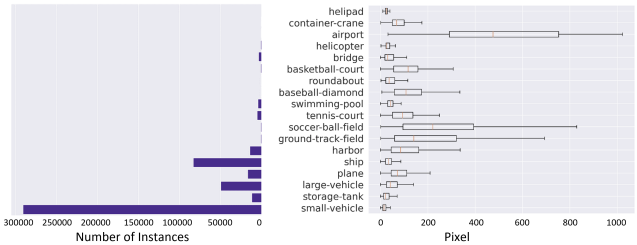
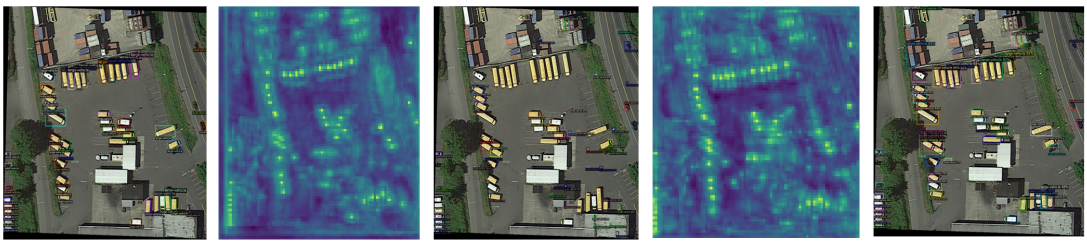


Fig. 6. DOTA [4] class and object size distribution.

IV. Experiments

The **DIOR** data set consists of 23,463 Google Earth images of areas in 80 countries. The quality of the images varies, and the content was captured during multiple seasons and multiple weather conditions. The data set covers a wide range of spatial resolutions, object size and object orientation variability, and a diverse class distribution, as illustrated in Fig. 5. The spatial resolution of the images is in the range $[0.5m, 30m]$, and the size of the images in the data set is 800×800 pixels. The number of annotated objects in the data set is 192,472, and they are categorized into 20 classes [5]. Our training set has 22,450, and the validation se has 1,012 images.

The **DOTA** data set consists of 2,430 overhead image images collected from Google Earth, and several other



(a) Ground Truth (b) Heatmap Base (c) Base Prediction (d) Heatmap SOD (e) SOD Prediction

Fig. 7. heatmap-based region proposal pipeline results for DOTA data.

satellites [4]. It is also known as the DOTA2.0 dataset, but we refer to it as DOTA in this paper for simplicity. Google Earth image sizes in the collection range from 800×800 to 4000×4000 pixels, the image size of the GF-2 satellite is $29,200 \times 27,620$ pixels. The GSD range in the DOTA dataset is 0.1 to 0.87 m, and the average number of objects per image is 220. The data set contains 1,793,658 annotated objects and is grouped into 18 classes. Most objects have a total size less than 50 pixels, and objects classes *small vehicle*, *ship*, *plane*, and *large vehicle* are densely packed in the images, as illustrated in Figure 6. In the experiment, we split the large images into subimages of size 1024×1024 pixels with an overlap of 200 pixels. Our training set has 12,700 images and the validation set has 4,543 images.

$$P = \frac{TP}{TP + FP} \quad R = \frac{TP}{TP + FN} \tag{3}$$

$$AP = \sum_{k=0}^{k=n-1} [R(k) - R(k+1)] * P(k) \tag{4}$$

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP(k) \tag{5}$$

Precision P, Recall R, Average Precision AP and mean AP over all classes are computed in Eq. 3, Eq. 4, and Eq. 5. True positives TP are results that the model predicted correctly, false positives FP are outcomes the model missed, and true negatives TN are outcomes the model erroneously predicted. All of these metrics were calculated on the basis of an IOU of 0.5: 0.95 and the number of proposals per image was set to 256. Precision P measures the fraction of relevant occurrences among recovered instances, and recall R is the fraction of objects that the model correctly identified among all relevant instances. $AP(k)$ is the Average Precision (AP) of class k in the test set and is calculated as the weighted sum of precision at each threshold (n is the number of thresholds), and the weight is the increase in recall (Eq. 4), and mAP is an average value of $AP(k)$ over n classes in the data set (Eq. 5).

Precision of two models in two different data sets is illustrated in Figure 8(a). The average precision of the *Base* model in the 18 classes is 49.7% in the DIOR training set. The average precision of the *Base* model in DOTA validation data set is 17.8%. *SOD* model shows a reasonable improvement with the integration of the improved backbone and the difficulty scoring module in Figure 8 in terms of precision and recall. Figure 8(a) shows an improvement in AP of 1.5% for DOTA and 2.5% for the DIOR data

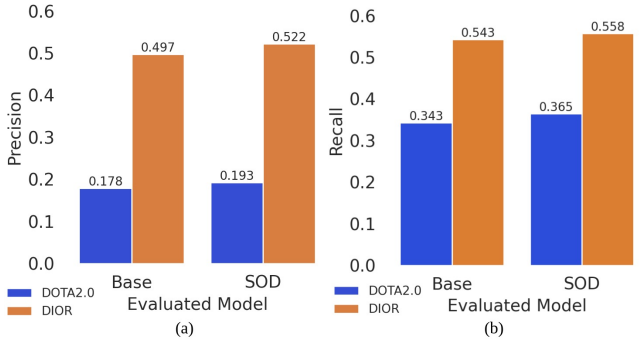


Fig. 8. (a) Precision(IOU=0.50:0.95) and (b) Recall(IOU=0.50:0.95) comparison from different models vs different datasets.

set. The improvement of the *SOD* model is significant for classes with small objects and classes that are difficult to distinguish, see Table I. **Recall** is illustrated in Figure 8(b). The recall with *Base* model for the DIOR and DOTA data sets is respectively 54.3% and 34.3%. The *SOD* model with improved backbone and difficulty module achieved a significant improvement in all data sets. The recall measure for the DIOR data set was 1.5% higher than the *Base* model (see Figure 8(b)) and increase in DOTA dataset is 2.2% compared to *Base* model. The **AP** for the small rare classes for two datasets and models is presented in Table I. The mAP for the DIOR data set is 49.6% for the *Base* model and 51.9% for the *SOD* model, as illustrated in Table I. The mAP for the DOTA dataset is 17.1% for the *Base* model and 18.9% for the *SOD* model. a 1.8% increase as in Table I.

V. Conclusion and Future Work

Object detection in aerial images is one of the most challenging tasks in computer vision research due to many small and overlapped objects in the images. The success of DNN object localization depends on the performance of the large number of related objects annotated in the training data and on a reliable feature extractor module in the pipeline. In this paper, we introduce a strong feature extractor that captures balanced low-level and high-level features for small objects. Next, we introduce the heatmap-based region proposal module to better capture small objects. Finally, we introduce two new modules in the satellite object detection pipeline, the difficulty scoring module, which informs the image difficulty to the custom focal loss module and balances the aerial object detector against trivial background classes. The proposed *SOD* method performed well (see Figures 4 and 7) on the DIOR and DOTA data set.

TABLE I
DIOR AND DOTA AP SCORES FOR SMALL AND DIFFICULT CLASSES

class label	mAP	Bridge	Service Area	**Harbor**	**Ship**	Storage Tank	**Track**	**Station**	Tennis Court	Overpass	Airplane	Dam	Airport	Toll Station
Num. Ann.	NA	207	67	259	2494	2629	154	58	580	163	844	33	56	67
Base	49.6	22.86	54.63	35.03	52.14	42.32	52.60	27.14	74.75	34.92	65.43	29.30	53.73	42.727
SOD	51.9	24.84	58.85	39.72	55.47	44.81	54.25	31.22	76.27	37.51	68.32	31.18	58.12	45.61

class label	mAP	Plane	Bridge	**Small V**	Large Vehicle	Ship	Basketball Court	Storage Tank	Roundabout	**Harbor**	Helicopter	**Crane**	Helipad	Airport
Num. Ann.	NA	3792	634	53660	6739	17650	240	3045	214	3689	86	28	4	89
Base	17.1	36.18	8.61	10.14	21.68	21.23	21.78	18.13	14.32	19.58	10.36	0.00	0.00	11.35
SOD	18.9	38.23	10.33	11.74	21.82	22.94	22.88	20.21	15.10	21.06	12.11	2.41	1.98	14.11

Our proposed model outperformed the baseline model in the very difficult DOTA satellite dataset by 1.5%, 2.2%, and 1.8% for precision, recall, and mAP metrics, respectively.

ACKNOWLEDGEMENT

This work is partially supported by the NAVAIR SBIR N68335-18-C-0199. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government paper references. The Titan Xp used for this research was donated by the NVIDIA Corporation.

REFERENCES

[1] Scott Workman and Nathan Jacobs. Dynamic traffic modeling from overhead imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12315–12324, 2020.

[2] Jia Liu, Jianjian Xiang, Yongjun Jin, Renhua Liu, Jining Yan, and Lizhe Wang. Boost precision agriculture with unmanned aerial vehicle remote sensing and edge intelligence: A survey. *Remote Sensing*, 13(21):4387, 2021.

[3] Payal Mittal, Raman Singh, and Akashdeep Sharma. Deep learning-based object detection in low-altitude uav datasets: A survey. *Image and Vision computing*, 104:104046, 2020.

[4] Jian Ding, Nan Xue, Gui-Song Xia, Xiang Bai, Wen Yang, Michael Yang, Serge Belongie, Jiebo Luo, Mihai Datcu, Marcello Pelillo, and Liangpei Zhang. Object detection in aerial images: A large-scale benchmark and challenges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021.

[5] Ke Li, Gang Wan, Gong Cheng, Liqiu Meng, and Junwei Han. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS Journal of Photogrammetry and Remote Sensing*, 159:296–307, 2020.

[6] Xiang Long, Kaipeng Deng, Guanzhong Wang, Yang Zhang, Qingqing Dang, Yuan Gao, Hui Shen, Jianguo Ren, Shumin Han, Errui Ding, et al. Pp-yolo: An effective and efficient implementation of object detector. *arXiv preprint arXiv:2007.12099*, 2020.

[7] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[8] Xingkui Zhu, Shuchang Lyu, Xu Wang, and Qi Zhao. Tph-yolov5: Improved yolov5 based on transformer prediction head for object detection on drone-captured scenarios. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2778–2788, 2021.

[9] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

[10] Pengfei Zhu, Longyin Wen, Xiao Bian, Haibin Ling, and Qinghua Hu. Vision meets drones: A challenge. *arXiv preprint arXiv:1804.07437*, 2018.

[11] Adam Van Etten. You only look twice: Rapid multi-scale object detection in satellite imagery. *arXiv preprint arXiv:1805.09512*, 2018.

[12] Delia-Georgiana Stuparu, Radu-Ioan Ciobanu, and Ciprian Dobre. Vehicle detection in overhead satellite images using a one-stage object detection model. *Sensors*, 20(22):6485, 2020.

[13] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Probabilistic two-stage detection. *arXiv preprint arXiv:2103.07461*, 2021.

[14] Xin Wu, Wei Li, Danfeng Hong, Ran Tao, and Qian Du. Deep learning for unmanned aerial vehicle-based object detection and tracking: a survey. *IEEE Geoscience and Remote Sensing Magazine*, 10(1):91–124, 2021.

[15] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.

[16] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[17] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020.

[18] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 390–391, 2020.

[19] Yang Liu, Peng Sun, Nickolas Wergeles, and Yi Shang. A survey and performance evaluation of deep learning methods for small object detection. *Expert Systems with Applications*, 172:114602, 2021.

[20] Du Jiang, Gongfa Li, Chong Tan, Li Huang, Ying Sun, and Jianyi Kong. Semantic segmentation for multiscale target based on object recognition using the improved faster-rcnn model. *Future Generation Computer Systems*, 123:94–104, 2021.

[21] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019.

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[23] Zuoxin Li and Fuqiang Zhou. Fssd: feature fusion single shot multibox detector. *arXiv preprint arXiv:1712.00960*, 2017.

[24] Lisha Cui, Rui Ma, Pei Lv, Xiaoheng Jiang, Zhimin Gao, Bing Zhou, and Mingliang Xu. Mdssd: multi-scale deconvolutional single shot detector for small objects. *arXiv preprint arXiv:1805.07009*, 2018.

[25] Jiaxu Leng, Yihui Ren, Wen Jiang, Xiaoding Sun, and Ye Wang. Realize your surroundings: Exploiting context information for small object detection. *Neurocomputing*, 433:287–299, 2021.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[27] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.