STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **13**, April 2025, pp 1536–1543.
Published online in International Academic Press (www.IAPress.org)

Aerial Remote Sensing Object Detection using Unsupervised Domain Adaptive

Youssef Ben Youssef [1,*], Soufiane Lyaqini [2], Khaled Fakhar [2], Elhassane Abdelmounim[1]

[1]*Department of Applied Physics,FST, Hassan First University Settat, Morocco*
[2]*Department of Mathematics and Computer Sciences Engineering, Hassan First University Settat, Morocco.*

Abstract Object recognition and localization in Aerial Remote Sensing Images (ARSI) are critical and demanding subjects for further processing object-related data in civil and military applications. To train a Deep Learning (DL) model for visual recognition and localization, a huge number of annotated images are needed. However, data categorization and annotation become a hard and time-consuming task. Despite the shortcomings of data in training, Unsupervised Domain Adaptation (UDA) offers an alternative solution to this issue. In this paper, UDA is suggested to detect and localize objects in ARSI as an unlabeled target domain. We compare the effectiveness of Faster Region Convolutional Neuronal Network (Faster R-CNN) as a two-stage detector and RetinaNet as one one-stage detector. These algorithms are based on the same Resnet50 model as the backbone. This study uses the natural image dataset MSCOCO as the source domain. We assess the proposed approach on two unlabeled datasets UC Merced and MTRASI datasets. The proposed method significantly improves object detection and localization performance, according to both qualitative and quantitative results. Extensive experiments show that the RetinaNet detector is better than the Faster R-CNN detector in terms of mAP.

Keywords Unsupervised Domain Adaptation, Convolutional Neuronal Network, Deep Learning, Object Detection, Aerial Remote Sensing Images

AMS 2010 subject classifications 97R40, 68T30

DOI: 10.19139/soic-2310-5070-1749

1. Introduction

A variety of DLs based on object detection and localization approaches have recently presented greatly improving object detection scores. One of the most prominent forms of DL is supervised learning, which has also demonstrated success in various application areas [1, 2]. Object identification is critical in image interpretation and is also significant in a variety of applications such as complex urban environments, precision agriculture, automated driving, and intelligent monitoring[3, 4]. To train a supervised DL model for object recognition and localization, a large number of labeled images need to be used. However, visual data labeling takes time, and hard to add the necessary information [5]. Researchers have made relentless efforts to solve the problems of detecting objects in ARSI based on CNN. As a result, UDA offers a solution to this problem, which we have provided to detect and localize objects in ARSI as unlabeled target domain [6, 7]. UDA refers to the field of study that leverages knowledge obtained in the source domain to solve a related but distinct domain target [8, 9]. Several civil and military applications based on ARSI object recognition provide more specific information that is used in various DL algorithms, such as airplane detection, airport security, and flight tracking [10]. UDA has been identified as a critical component of machine learning used in a wide range of fields [11]. In the literature,

*Correspondence to: Youssef Ben Youssef (Email: youssef.benyoussef@uhp.ac.ma). Department of Applied physics, Hassan First University.Km 3, B.P.: 577 Route de Casablanca(26000), Morocco.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

deep object detectors are classified into two types: i) Single-stage approaches predict bounding boxes around objects and class probabilities for example RetinaNet [12]; ii) Two stages use a selective search to find probable object locations based on image attributes and then classify them using Convolutional Neural Network (CNN) for example Faster R-CNN [13]. Object detector algorithms attempt to identify and classify one or more objects in an image. Domain adaptive Faster R-CNN is a system based on both regression and classification, and it contains multiple components including a feature extractor, and region proposal network. In contrast, domain-adaptive RetinaNet uses a unified network composed of a backbone network and two subnetworks. The first subnet uses convolutional object classification to classify the output of a backbone, while the second uses convolutional bounding box regression to detect object candidates. A feature pyramid network is connected to its backbone to generate multi-scale pyramid features. These detectors have gained interest in a variety of applications, including robots, self-driving cars, and autonomous surveillance[14, 15]. The Domain Adaptive Faster R-CNN model was presented to detect aircraft from remote sensing images in [16]. The authors have proposed adversarial training to facilitate domain shift. Han et al. proposed a detection algorithm in remote sensing images that embeds Mask R-CNN with traditional object detection algorithms [17]. Zhou Liming et al. proposed the Multiscale Detection Network to address the issue of small aircraft shape, while also proposing the Deeper and Wider Module to overcome background noise [18]. Tahir et al. proposed a method for aircraft detection systems based on the YOLO deep learning model to address the problem of object detection in satellite images, which was complex because objects had many variations, types, poses, sizes, and complex backgrounds [19]. Wei et al. proposed the X-LineNet-based single-stage aircraft detector. The network can learn through visual grammar thanks to the model, which transforms the objective of aircraft detection in remote sensing images from detection to prediction [20]. Ju et al. used the RetinaNet detector to detect landslides in remote sensing images [21]. A classifier aircraft in remote sensing images based on deep CNN is presented in our previous works [23, 22]. In ARSI, building from scratch an object detector based on deep learning is labor consumption since several annotated images and instances of objects are required. Thus, we propose UDA-based object detectors to solve this issue. The main contributions to this work are:

- We adopt unsupervised domain adaptive to detect and localize objects in ARSI.
- We demonstrate that the RetinaNet detector based on one stage is better than the Faster R-CNN detector based on two stages.

The findings show considerable promise in item recognition in several real-world contexts. UDA increases the efficiency, accuracy, and cost-effectiveness of technologies used in surveillance, disaster response, and other fields by allowing them to adapt to new and diverse conditions without the need for vast amounts of labeled data. The following is the rest of this work: Section 2 introduces the basic theory of UDA, while Section 3 details the proposed method. Experiments and results are shown in Section 4. The conclusion is drawn in Section 5.

2. Domain Adaptive Basic Theory

We will provide some fundamental symbols that briefly describe the notion of UDA. The source domain is considered to have all available labels, and it may be specified as follows: $D_s = \{x_s^i, y_s^i\}_{i=1}^{N_s}$ where N_s is the number of labeled images and y_s^i denotes object labels in the corresponding image and the target domain $D_t = \{x_t^i\}_{i=1}^{N_t}$ where N_t is the number of target example x_t^i without labels. The target domain and source domain are derived from various probability distributions. Source domain has four parts: the feature space $\mathcal{X}$, the label space $\mathcal{Y}$, the marginal probability distribution $P(x)$ and the conditional probability distribution $P(x/y)$, where $\mathbf{X} \in \mathcal{X}$, $\mathbf{Y} \in \mathcal{Y}$. Because the source and target domains are distinct, their probability distributions differ $P(X_s^i) \neq Q(X_t^i)$. UDA intends to build an h(.) model which can detect object x in the target domain:

$$\hat{y} = h(x) \tag{1}$$

where $\hat{y}$ is the prediction. Thus, UDA is aimed to minimize the target risk $\varepsilon_t(h)$ using data source domain [24]:

$$\varepsilon_t(h) = \underset{(x,y)\sim Q}{Pr}[h(x) \neq y] \tag{2}$$

The detector based on deep learning aims to learn from domain source data by minimizing the loss function. The Faster R-CNN detector's loss function includes two different loss functions: classification and box regression [13]:

$$\mathcal{L}(p_i, t_i) = \mathcal{L}_{cls} + \mathcal{L}_{reg} = \frac{1}{N_{cls}} \sum_i (p_i^* \log(p_i) - (1 - p_i^*) \log(1 - p_i) + \frac{\lambda}{N_{box}} \sum_i p_i^* L_1^{smooth}(t_i - t_i^*) \tag{3}$$

where p_i is the predicted probability that box i is an object, p_i^* is the ground truth label indicating whether box i is an object. t_i is predicted four parameterized coordinates and t_i^* is coordinates of ground truth. λ is a balancing parameter, set to 10. $\mathrm{L}_1^{\mathrm{smooth}}$ is the smooth L_1 loss. The RetinaNet based on one stage detector is trained in the source domain by minimizing loss function called focal loss [12] :

$$FL(p_t) = -\alpha(1 - p_t)^{\gamma}. \log(p_t) \tag{4}$$

where p_t is the estimated probability for the ground truth class. and $(1 - p_t)^{\gamma}$ for scaling the loss function. α and γ are the parameters to control the importance weight. The focal loss function is constructed on the standard cross-entropy loss function. We used $\alpha = 0.25$ and $\gamma = 2$ for more flexibility over the design of the weighting function.

3. Proposed Method

Our methodology consists of aligning the distribution between the source and target domain, among the methods used in UDA[25]. We use Faster R-CNN and RetinaNet models with the same backbone Resnet50 [26] to ensure a fair comparison. The MSCOCO dataset is the preferred dataset and benchmark for object detection because of its thorough diversity and comprehensiveness [27]. Microsoft provides the MSCOCO dataset, which currently ranks among the best publicly available object detection datasets. The choice of the MSCOCO dataset as the source domain aligns well with the characteristics of ARSI, providing a diverse set of object classes, annotations in complex urban environments, and relevance to applications like precision agriculture, automated driving, and intelligent monitoring. It contains 300,000 segmented images of 80 different object categories, including accurate position labels. On average, each image contains about 7 objects, and instances appear at very large scales. Regardless of how useful this dataset is, object types other than the 80 selected classes will not be detected if trained solely on it. Figure 1 depicts an overview of the proposed method using Faster R-CNN and RetinaNet detectors.
To verify the effectiveness of our method, we use two datasets UC Merced and MTRASI, as target domains in our experiments. The MTRASI dataset contains 1945 optical remote sensing images collected from Google Earth. In total, 20 different types of aircraft are located in various airports worldwide. The size of all these ARSI is 256×256 and the spatial resolution is about 0.3m to 1.0m. This dataset also contains multi-temporal and orientation information and has only the annotation of the category [28]. UC Merced is a land-use remote sensing image collection with 21 classes and 100 images per class with size 256×256 and 30cm pixel resolution. Images were extracted manually from the national map urban area imagery collection for numerous cities around the USA [29]. The common feature of these two datasets is that the objects are unlabeled, which allows their inclusion in this work. Figure 2 depicts the two dataset samples.

4. Experimental results and Discussion

All experiments are implemented on PyTorch[30], detecto[31] and Keras [32]. The backbone used in our experiments is publicly available. All of our networks were built with PyTorch and Keras in the Google Colaboratory cloud environment, with the following hyper-parameters set: epochs=10, learning rate=0.001, batch size=28, and SGD optimizer. Experiments are conducted using a RetinaNet implementation based on FPN combined with a pre-trained ResNet50 backbone. Based on the output categories of the test samples compared to the output categories of the true labels, there are four possible outcomes: false positives (FP), false negatives

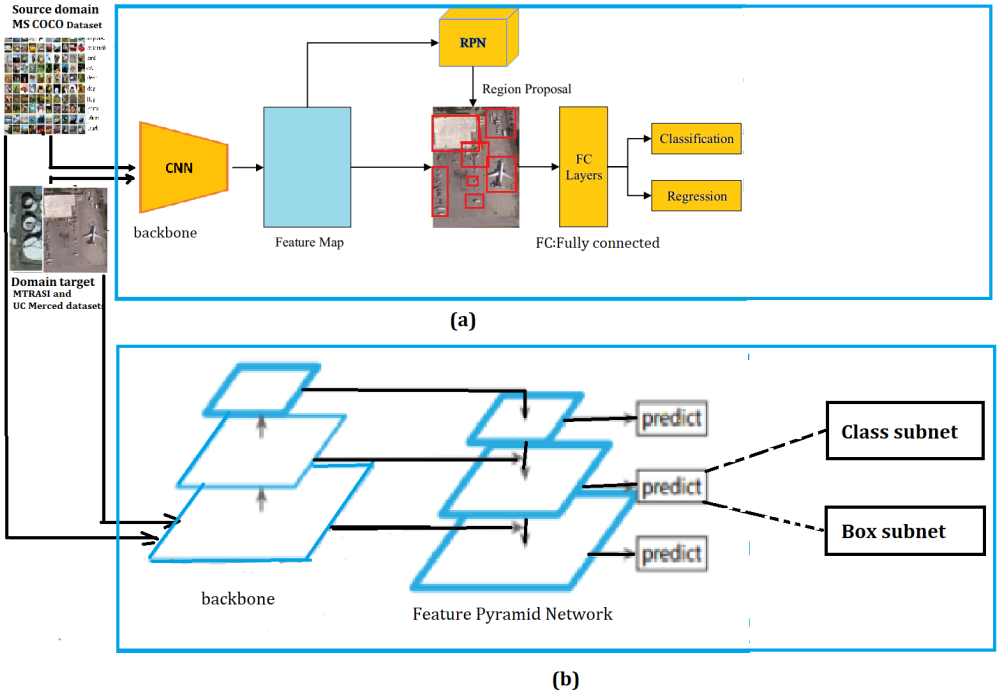

Figure 1. Overview of the proposed method: Faster R-CNN detector (a), RetinaNet detector (b).

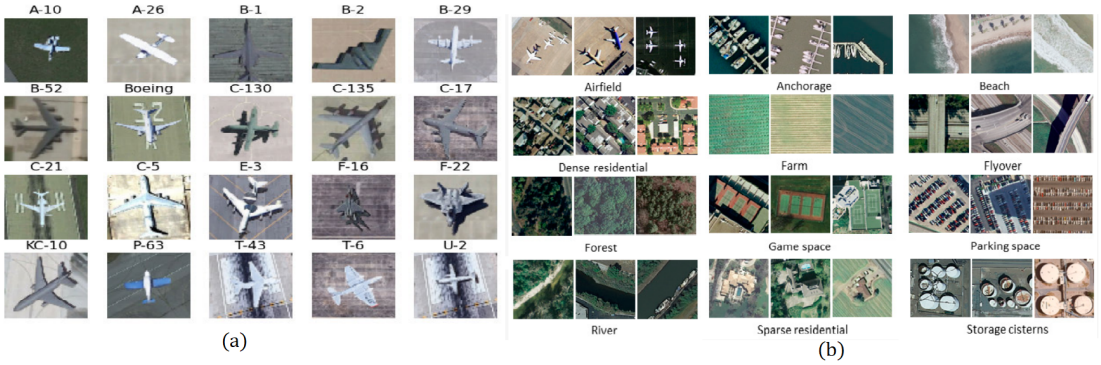

Figure 2. Samples of image extracted from target domains: MTRASI dataset(a), UC Merced dataset(b).

(FN), and true negatives (TN). All reported results follow the standard COCO mAP metric (score), which combines the Intersection over Union (IoU) with different thresholds over 10 IoU thresholds between 0.5 and 0.95 with a size of 0.05. Precision, recall, and score are defined as:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{5}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{score} = \frac{1}{N_c} \sum_{class} \frac{TP}{TP + FP} \tag{7}$$

Table 1. The score achieved with Faster R-CNN on 10 categories. The best result is highlight with bold

	cell phone	airplane	bird	bench	chair	car	personne	train	backpack	trafic light	mAP(%)
source → MTRASI	19.96	86.61	45.33	24.80	8.96	48.24	**99.27**	71.82	5.48	8.31	41.87
source → UCM	2.1	43.71	45.33	8.1	**44.08**	28.55	22.42	40.48	11.34	28,49	27.46

where N_c is total of classes. Table 1 summarizes experimental results in term of score on 10 categories.

From Table 1, the score has aggressively increased from the UC Merced dataset to the MTRASI dataset by 14.41%. The Faster R-CNN detector achieves the best score (41,87%) for the MTRASI dataset. Figures 3 and 4 show the visualization of object detection results: As can be seen from Figures 3 and 4, the qualitative detection

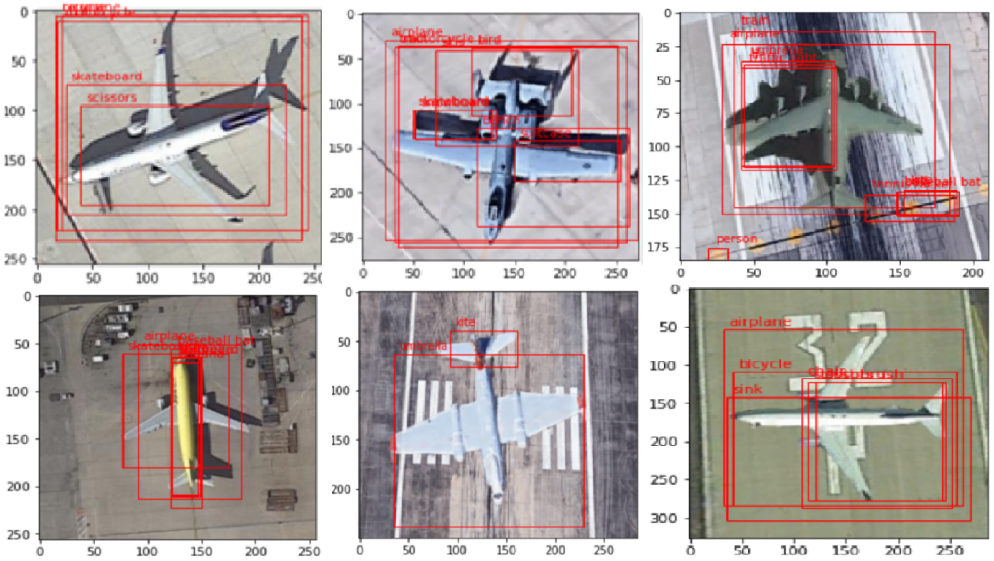

Figure 3. Object detection samples on the MTRASI dataset using Faster RCNN detector.

of our detector based on Faster R-CNN detects more objects with small size in both datasets. The results are summarized in Table 2 when using the RetinaNet detector. 10 objects detected are presented with their scores

Table 2. The score achieved with RetinaNet detector on 10 categories. The best result is highlighted with bold

	scissors	bird	airplane	parking meter	motorcycle	train	brocolli	clock	binch	car	mAP(%)
source→ MTRASI	53.23	11.27	**63,66**	23.11	61.65	53.17	19.43	13.11	21.81	45.72	36.61
source→ UCM	13.76	56.54	58.15	**68.22**	34.87	11.32	51.73	53.31	30.23	47.25	42.53

in Table 2. The RetinaNet detector achieves the highest score (42,53%) for the UC Merced dataset. The score increased from UC Merced dataset to MTRASI dataset by 5.92%. Other objects with different average precision are detected despite their absence in the domain target due to the large number of images used in the training and testing phases in the domain source, and some objects are only a few pixels in size due to resolution constraints caused by limited object information when the images are acquired. We notice there is a gap between the results obtained with the two datasets. Figures 5 and 6 show samples of the visual qualitative detection results. Despite a series of missed objects, we find that the RetinaNet detector has a better score than the Faster R-CNN detector. The score increased by 2.5% from MTRASI to UC Merced. Tables 1 and 2 demonstrate that RetinaNet is unable to detect some objects that the faster R-CNN can. The main explanation is that their architectures are quite different. We chose a common car category to compare our method to others because researchers use various datasets. Table 3 reports the best score of our approach achieved by Faster R-CNN in the MATRASI target domain compared with other works on the common car category. Table 3 shows that our method performs about 1.64% better than the most

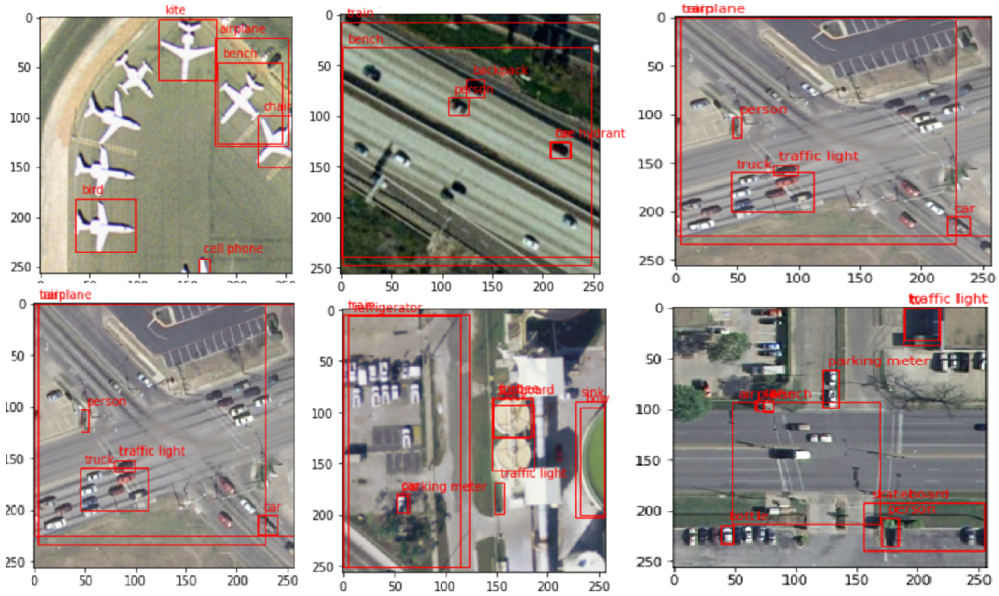

Figure 4. Object detection samples on UC Merced dataset using Faster CNN detector.

1.

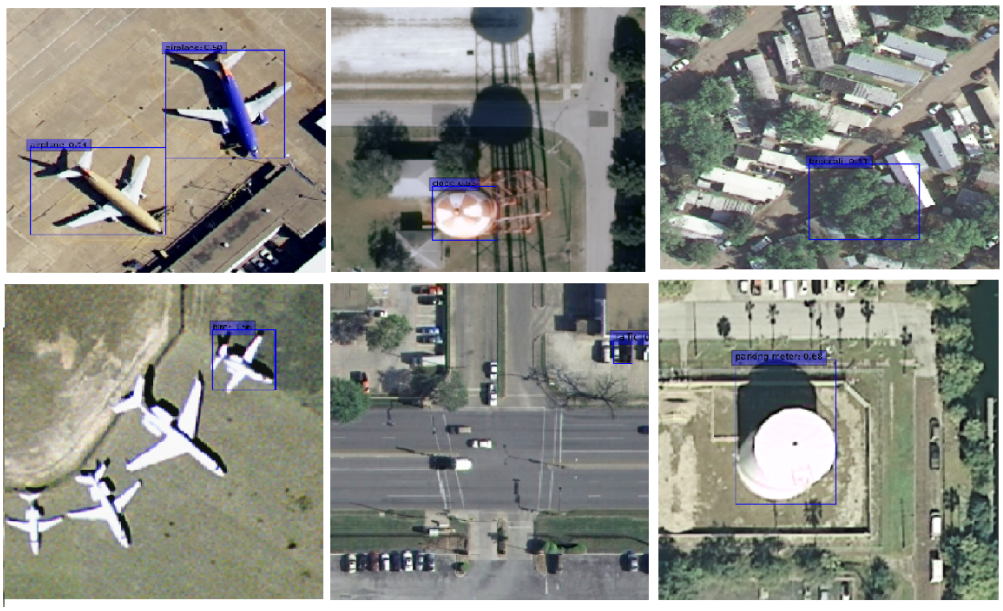

Figure 5. Examples of the qualitative results on the UCM dataset using the RetinaNet detector.

Table 3. Comparison Result of the proposed method.

	car score(%)
Ref. [33]	46.6
Ref. [34]	41.87
Ref. [35]	43.4
Our method	48.24

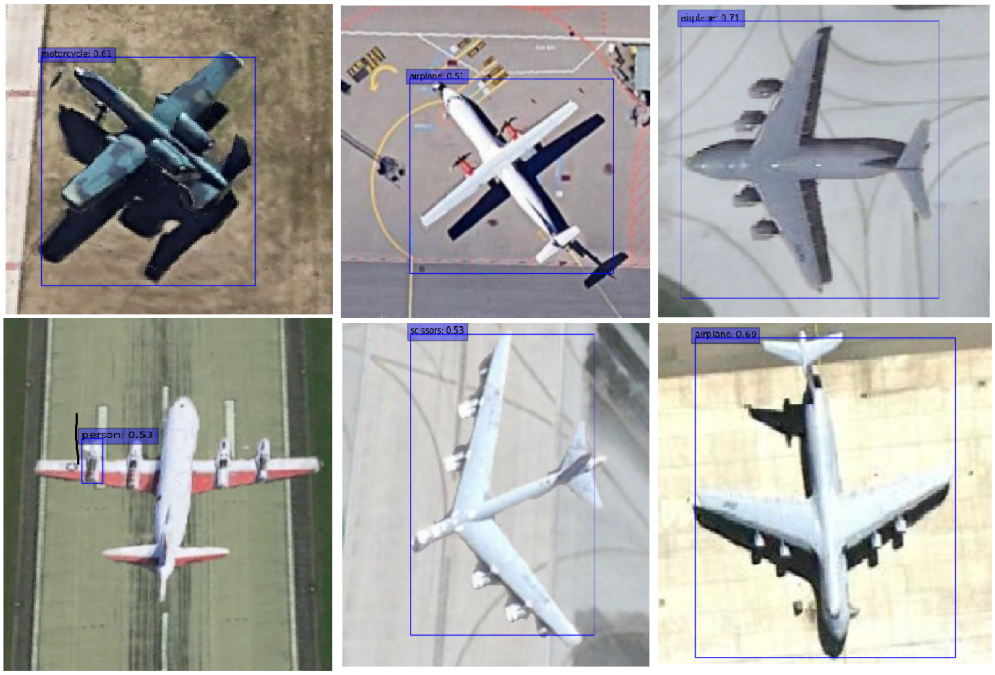

Figure 6. Examples of the qualitative results on the MTRASI dataset using the RetinaNet detector.

recent methods. These findings highlight how crucial feature alignment is when shifting one domain into another. However, our method also produced unsatisfactory results because it was unable to detect all objects because of their less obvious features.

5. Conclusion

This study proposes an unsupervised domain adaptive approach for detecting and localizing objects in aerial remote-sensing images without annotation. Two publicly available datasets are used to compare the Faster R-CNN and RetinaNet detectors. We demonstrate that those models achieve state-of-the-art performance. According to the score used for evaluation, the Faster R-CNN-based two-stage detectors have a lower score, whereas the RetinaNet one-stage detector has a higher score, even though some objects are missed. Despite UDA being a very effective object detection method, it has flaws. Integrating multi-source and multi-target domain adaptive algorithms is a promising field for future research in UDA for object recognition. This integration has the potential to significantly increase detection scores, robustness, and flexibility in object detection systems, increasing their success in a variety of real-world applications. By overcoming the difficulties and studying the available research fields, the discipline can advance toward more robust and adaptable AI systems.

REFERENCES

1. S. Amirian, Z. Wang, T.R. Taha and H.R. Arabnia, *Dissection of Deep Learning with Applications in Image Recognition*, In International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, pp.1142–1148,2018.
2. G. Wilson and D.J. Cook, *A survey of unsupervised deep domain adaptation*, ACM Transactions on Intelligent Systems and Technology (TIST),vol. 11, no 5, pp. 1–46,2020.
3. L. Liu et al. *Deep learning for generic object detection: A survey*, International journal of computer vision,vol.128, no 2, pp.261–318, 2020.
4. S. Zhao et al. *A review of single-source deep unsupervised visual domain adaptation*, IEEE Transactions on Neural Networks and Learning Systems (2020).

5. S.J. Pan and Q. Yang, *A survey on transfer learning*, IEEE Transactions on knowledge and data engineering,vol.22, no.10, pp.1345–1359,2010.
6. Oza, Poojan, et al. *Unsupervised domain adaptation of object detectors: A survey*, arXiv preprint arXiv: 2105.13502,2021.
7. Q.Z. Zhong, X. Shou-tao, and W. Xindong, *Object Detection with Deep Learning: A Review*, IEEE Transactions on Neural Networks and Learning Systems (2019).
8. G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, *When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs*, in IEEE Transactions on Geoscience and Remote Sensing, vol. 56, no.5, pp. 2811–2821, 2018.
9. L. Zhang, M. Lan, J. Zhang, and D. Tao, *Stage wise Unsupervised Domain Adaptation with Adversarial Self-Training for Road Segmentation of Remote-Sensing Images*, IEEE Transactions on Geoscience and Remote Sensing, vol.60, pp. 1–13,2021.
10. L. Jiao et al. *A survey of deep learning-based object detection*, IEEE Access, vol.7, pp.128837–128868,2019.
11. L. Andersson, M. Lupu, and A. Hanbury, *Domain adaptation of general natural language processing tools for a patent claim visualization system*, In Information Retrieval Facility Conference Springer, Berlin, Heidelberg pp.70-82,2013.
12. T.Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, *Focal loss for dense object detection*, IEEE Trans. Pattern Anal. Mach. Intell, vol.42, pp. 318–327, 2020.
13. S. Ren, K. Hen, R. Girshick, and J. Sun, *Faster R-CNN: Towards real-time object detection with region proposal networks*, In Proceedings of the 28th International Conference on Neural Information Processing Systems, pp. 91–99,2015.
14. A. Gupta et al. *Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues*, Array vol.10, pp.100057, 2021.
15. L.B. Das et al. *Human Target Search and Detection using Autonomous UAV and Deep learning*, IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), Bali, Indonesia, pp.55–61, 2020.
16. J. Chen, J. Sun, Y. Li and C. Hou, *Object detection in remote sensing images based on deep transfer learning*, Multimedia Tools and Applications, vol.81, no.9, pp.12093–12109, 2022.
17. Q. Han, Q. Yin, X. Zheng, and Z. Chen, *Remote sensing image building detection method based on Mask R-CNN*, Complex Intelligent Systems,vol.8, no.3, pp.1847–1855,2022.
18. L. Zhou, H. Yan, Y. Shan, C. Zheng, Y. Liu, X. Zuo, B. Qiao, and Y. Li, *Aircraft Detection for Remote Sensing Images Based on Deep Convolutional Neural Networks*, Journal of Electrical and Computer Engineering 2021.
19. A. Tahir, M. Adil, and A. Ali, *Rapid detection of aircrafts in satellite imagery based on deep neural networks*, arXiv preprint arXiv:2104.11677,2012.
20. H. Wei, Y. Zhang,B. Wang, Y. Yang, H. Li, and H. Wang, *X-LineNet: Detecting Aircraft in Remote Sensing Images by a Pair of Intersecting Line Segments*, IEEE Transactions on Geoscience and Remote Sensing, vol.59, no.2, pp.1645–1659,2020.
21. Y. Ju, Q. Xu, S. Jin, W. Li, Y. Su, X. Dong, Q. Guo, *Loess landslide detection using object detection algorithms in northwest China*, Remote Sensing, vol.14, no.5 pp. 1182, 2022.
22. Y. Ben Youssef, M. Merrouchi,E. Abdelmounim, T. Gadi, *Classification of Aircraft in Remote Sensing Images using Deep Convolutional Networks* Statistics, Optimization and Information Computing, vol.10, pp.4–11,2022.
23. Y. Ben Youssef, M. Merrouchi,E. Abdelmounim, T. Gadi, *Aircraft Type Classification in Remote Sensing Images using Deep Learning* 2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS), Kenitra, Morocco, 2020, pp. 1-6, doi: 10.1109/ICECOCS50124.2020.9314611..
24. Ben-David et al. *A theory of learning from different domains*, journal Machine learning, Springer, vol.79,no.1,pp.151–175,2010.
25. G. Csurka, *A comprehensive survey on domain adaptation for visual applications*, In Csurka,G. (eds) Domain Adaptation in Computer Vision Applications. Advances in Computer Vision and Pattern Recognition. Springer, Cham, pp.1–35,2017.
26. K. HE, X. ZHANG, S. REN, et al. *Deep residual learning for image recognition*, In Proceedings of the IEEE conference on computer vision and pattern recognition, pp.770–778. 2016
27. T.Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona,D. Ramanan,and C.L. Zitnick, *Microsoft coco: Common objects in context*, In European conference on computer vision, Springer,Cham, pp. 740–755,2014.
28. Z.Z. Wu, S.H. Wan, and X. Fang, *A benchmark data set for aircraft type recognition from remote sensing images*, Applied Soft Computing Journal, vol.89, pp. 106–132, 2020.
29. Yi. Yang and N. Shawn, *Bag-of-visual-words and spatial extensions for land-use classification*, in Proceedings of the 18th SIGSPATIAL international conference on advances in geographic information systems. 2010.
30. A. Paszke et al. *Pytorch: An imperative style, high-performance deep learning library*, In Advances in neural information processing systems,vol.32,2019.
31. *Detecto Python package*, Available via: http://pypi.org/project/detecto/1.1.2 accessed on 30-10-2022.
32. S. Humbarwadi, *Object Detection with RetinaNet*, Available in https://colab.research.google.com/github/kerasteam/keras-io/blob/master/examples/vision/ipynb/retinanet.ipynb accessed on 15-10-2022.
33. Q. Cai et al. *Exploring object relation in mean teacher for cross-domain detection*, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.11457–11466, 2019.
34. M. Xu et al. *Cross-domain detection via graph-induced prototype alignment*, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.
35. Y. Chen, W. Li, C. Sakaridis, D. Dai, and L.V. Gool, *Domain adaptive faster R-CNN for object detection in the wild*, In IEEE Conference on Computer Vision and Pattern Recognition, 2018