IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 10, 2020, accepted October 18, 2020, date of publication October 22, 2020, date of current version November 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3032955

Real-Time Target Detection in Visual Sensing Environments Using Deep Transfer Learning and Improved Anchor Box Generation

ZHENBO REN[1,2], EDMUND Y. LAM[3], (Fellow, IEEE), AND JIANLIN ZHAO[1,2]

[1]MOE Key Laboratory of Material Physics and Chemistry under Extraordinary Conditions, Northwestern Polytechnical University, Xi'an 710129, China
[2]Shaanxi Key Laboratory of Optical Information Technology, School of Physical Science and Technology, Northwestern Polytechnical University, Xi'an 710129, China
[3]Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong

Corresponding authors: Zhenbo Ren (zbren@nwpu.edu.cn) and Jianlin Zhao (jlzhao@nwpu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61905197, and in part by the Fundamental Research Funds for the Central Universities under Grant 310201911qd002.

ABSTRACT Visual perception is critical and essential to understand phenomenon and environments of the world. Pervasively configured devices like cameras are key in dynamic status monitoring, object detection and recognition. As such, visual sensor environments using one single or multiple cameras must deal with a huge amount of high-resolution images, videos or other multimedia. In this paper, to promote smart advancement and fast detection of visual environments, we propose a deep transfer learning strategy for real-time target detection for situations where acquiring large-scale data is complicated and challenging. By employing the concept of transfer learning and pre-training the network with established datasets, apart from the outstanding performance in target localization and recognition can be achieved, time consumption of training a deep model is also significantly reduced. Besides, the original clustering method, k-means, in the You Only Look Once (YOLOv3) detection model is sensitive to the initial cluster centers when estimating the initial width and height of the predicted bounding boxes, thereby processing large-scale data is extremely time-consuming. To handle such problems, an improved clustering method, mini batch k-means++ is incorporated into the detection model to improve the clustering accuracy. We examine the sustainable outperformance in three typical applications, digital pathology, smart agriculture and remote sensing, in vision-based sensing environments.

INDEX TERMS Clustering methods, machine learning algorithms, machine vision, object detection.

I. INTRODUCTION

Vision is a significant and basic way to acquire information and explore the essence of the real world. In recent years, due to the rapid development of imaging devices in manufacturing, cameras are becoming cheaper and smaller, while maintaining higher capturing speed and resolution. As such, visual sensors are ubiquitously used in environments such as transportation systems [1], medical imaging [2], system status monitoring [3] as well as consumer products [4].

Despite the great promise offered by smart visual sensing devices, there are many challenges in realizing the opportunities. One of the key problems stems from the requirements of the digital system in real-time performance in dynamic environments, which is mainly caused by the large-scale data collected by numerous sensors. Such issue is particularly severe in vision-based projects, in which several high-speed high-resolution cameras are configured. Concretely speaking, vision-based sensing can significantly impact the health-care field by connecting distributed medical devices and adding intelligent modules for information acquisition and processing. In hospitals, human or technological errors caused by false alarms, slow response, and inaccurate information are still major reasons of preventable death and patient suffering. With the help of automatic imaging and analysis platform, for example, clinical testing like molecular imaging and subsequent specific cell detection/counting, cancer histology and MRI image reconstruction, can be precisely

The associate editor coordinating the review of this manuscript and approving it for publication was Jinjia Zhou.

performed in a short time. Accordingly, hospitals can significantly accelerate the testing speed and reduce uncertainty, thereby overcoming such limitations and improving patient safety and doctor experiences [5]. Also, in smart agriculture, by the use of cameras and processing platforms, the target crop/plant within a field at any time can be imaged. Farmers can then measure agricultural variables to get the knowledge of the status of soil, crop yield, and weed/pest control instantly, thereby raising productivity and facilitating precision agriculture [6]. Combining image analytics from sensing devices and advanced algorithms, smart agriculture can be successfully achieved. And similarly, powerful visual sensors also provide opportunities and possibilities to enhance efficiency, safety, and working conditions in field investigation. Using unmanned aerial vehicles (UAVs), drones or satellite embarked a camera in field monitoring allows people to inspect/gather geological information or surface features in the actual situation, or undertake daily land surveys by checking the status of oil tanks/pipelines and alarming/locating the wildfire in a no man's land or distant forest, thereby minimizing people' exposure to wild and hazardous zones in industrial and remote environments [7]. These vision-based applications are developing towards smart and digital processing, in which a camera is configured to capture high-resolution images or videos at a high relatively frames per second (FPS). As such, fast processing and analysis when handling massive data is necessary.

In recent years, driven by big data and computational capability, deep learning has become a powerful tool that deeply revolutionizes numerous areas such as computer vision and coherent imaging [8]–[10]. The use of computational intelligence in vision-based applications, especially in the above three topics, health-care, agriculture and field monitoring, has been undertaken for years. To be specific, in order to assist doctors in evaluating more patients and speed up the diagnostic process which in turn can reduce the time gap for treatments, deep learning is employed to automatically detect specific cells/tissues for screening process. Google AI team proposes to train a deep neural network to detect referable diabetic and reaches a higher F-score than professional ophthalmologists [11]. For the histological analysis at cellular and tissue level, authors of Ref. [12] train a deep learning model for detection and classification of colon cancer, thereby benefiting understanding of the tumor microenvironment. In Ref. [13], authors apply deep learning and end-to-end strategy on mammographic images to improve breast cancer detection. Sarraf *et al.* employ LeNet-5 to detect Alzheimer's disease in fMRI data and the network reaches a mean accuracy of 96.8% [14]. In smart agriculture, research efforts of deep learning have also been demonstrated. With cameras embarked on a UAV, images are extensively collected for tasks of fruit counting [15], weed detection and mapping [16], plant disease recognition [17] and plant identification [18]. In field monitoring which is empowered by remote sensing, Ref. [19] reports land-use and land-cover classification and reaches accuracies of 93.57% to 96.17% for three CNN-based algorithms, and Ref. [20] reports an F-score of 0.96 for a deep recurrent neural network. Besides, to better observe the earth surface and recognize aerial scenes acquired by remote imaging systems like satellite, drone and multi-spectrum imagery, researchers have made efforts on various topics such as airport [21] detection, natural hazards monitoring [22], as well as crop yield and vegetation detection [23] etc.

In this paper, to reduce the computational load and accuracy of generating anchor boxes for customized data, and to improve the performance and inference speed of target localization and recognition, with the help of advanced deep learning method, we propose a mini batch k-means++ method and a transfer learning strategy for real-time object detection using the You Only Look Once (YOLOv3) model. By employing heuristic initialization and mini batch clustering, anchor boxes are created in a smart and fast way. And by training a detection model with natural images and by reusing pre-trained weights, knowledge learned from one domain can be effectively transfered to a new domain. As such, burdens of generating clustering centers, collecting large-scale professional datasets and annotating each image manually are significantly ameliorated. Apart from the outstanding performance benefited from training with massive data, the time consumption of training models for specific vision-driven projects is also greatly reduced. We demonstrate the capability of the proposed scheme by applying to three typical vision-based sensing environments: cell detection in digital pathology, crop detection in smart agriculture and aerial scene detection in earth observation and remote sensing. Code snippets can be found at https://github.com/thomas0708/object-detection.

The contributions of this paper can be summarized as follows

1) This work shows that by using an improved clustering strategy, computational burden of generating new anchor boxes for customized data is significantly ameliorated and the quality of new cluster centers is improved.
2) Deep transfer learning can significantly relax the time-consuming training process by pre-training the model with natural images and by transferring weights to small-scale customized datasets in professional communities.
3) The superior performance and real-time detection of using the improved clustering method and transfer learning are verified by applying to three detection cases: digital pathology, smart agriculture and remote sensing.

This paper is organized as follows. Basic principle of object detection is reviewed and the proposed training strategy, transfer learning, and the improved anchor box generation method are explained in Section II. Comparison of the proposed method and the experimental results of three vision-based detection scenarios are introduced in Section III. Finally, Section IV addresses the concluding remarks.

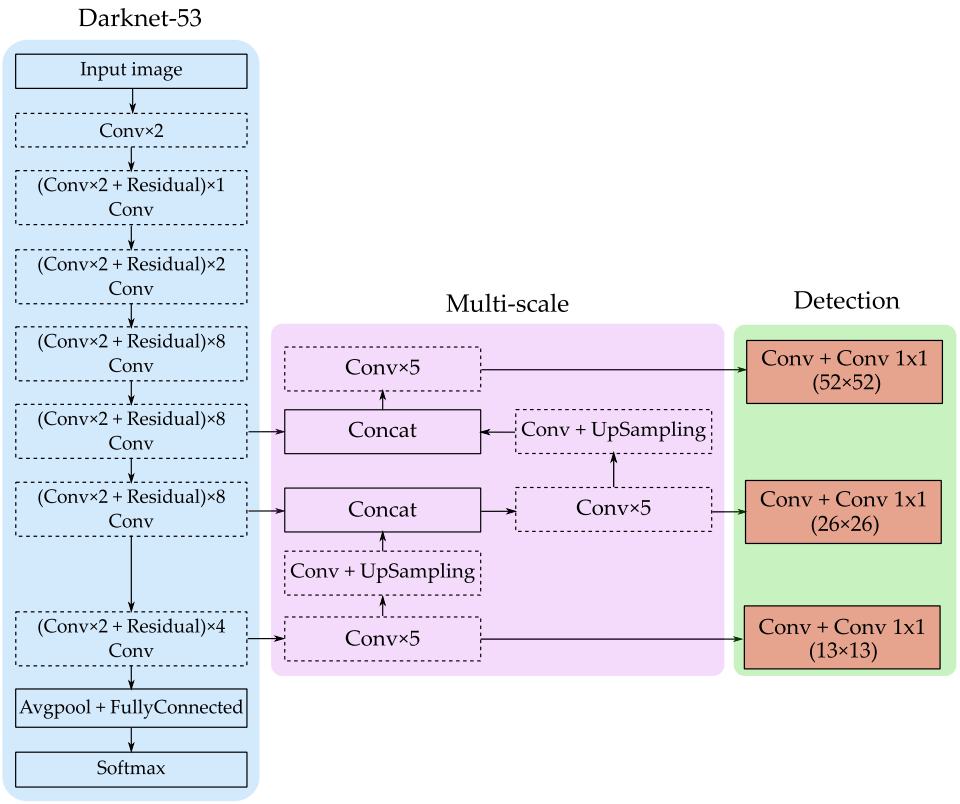


FIGURE 1. Pipeline of detection network using YOLOv3 model. In this architecture, "Conv" denotes the 2D convolution, "Residual" is the skip connection, "Avgpool" is the average pooling, "Concat" denotes the concatenation, "×*n*" in blocks means that the operation is repeated *n* times, (*N* × *N*) in *Detection* shows different scales of feature maps.

II. PRINCIPLE AND METHODS

A. DEEP LEARNING AND OBJECT DETECTION

In recent years, deep learning, or in other words, convolutional neural network, has received increasing attention from the industry and academia. Thanks to its outstanding performance in computer vision and natural language processing, it has been successfully applied to numerous real-world applications [24]. Generally speaking, the deep learning algorithm aims to automatically learn high-level features from massive data, making it beyond traditional machine learning where features have to be manually and deliberately designed. By supervised, unsupervised or semi-supervised learning, representation features can be extracted with hierarchically cascading functional layers, including convolutional layer, pooling layer, fully connected layer and activation layer etc [24]. Usually a deep network is constructed with multiple layers since such network has more fitting variables and can enrich the representation learning capability from data.

Object detection is more challenging compared to classification. Not only multiple objects in a single image need to be correctly recognized (*recognition*), but also their individual locations are required to be detected (*localization*) [25]. Popular detection algorithms empowered by deep learning can be categorized into two groups [26]. The first type is based on region proposal CNN, including R-CNN and its derivatives like Fast R-CNN, Faster R-CNN, Mask R-CNN etc. They follow traditional detection pipeline by firstly generating region proposals and then classifying each proposal into respective target categories. The second group regards the detection task as a regression or classification problem. A single framework is adopted to predict the final category and location of a target directly, resulting in the so-called one-stage detection. Typical models of one-stage detection are AttentionNet, Single Shot MultiBox Detector (SSD), and YOLO [27]. Considering the requirement of prediction speed in vision-based object detection applications in the industry, we select the third version of YOLO, YOLOv3 [28], [29], as the detection model. The full network is shown in Fig. 1.

In this model, there are three modules, *Darknet-53*, *Multi-scale* and *Detection*. A deep CNN, Darknet-53, is built as the backbone for feature extraction. A key point of YOLOv3 is the use of three scale feature maps at the output layer. They are designed for multi-scale detection, enabling the network to recognize small targets. At the end of detection, given all the scored regions in an image, non-maximal suppression is implemented to retain the winner bounding box and class. The loss function of YOLOv3 is the sum of the mean square error of coordinate error, intersection over union (IoU) error and classification error [27]. Adam optimizer is used to minimize the loss function while training the network. In details of method implementation, in which the IoU threshold is set as 0.4 and confidence score threshold is 0.35. We set the learning rate empirically to 0.0001, and make

it decay exponentially with a rate of 0.9 as the training progresses every 10 epochs. The training epoch is 300, and each mini-batch contains 16 images. All images are resized to 416 × 416 before training and testing. After prediction, the image as well as the detected bounding box is resized back to its original size. We implement the model using TensorFlow and Keras and all the experiments are performed with a CPU of Intel Core i7@3.6GHz and a GPU of Nvidia Titan RTX.

B. DEEP TRANSFER LEARNING

Despite of its power in many vision-based applications, unfortunately, in order to get the knowledge of the latent patterns and mathematical distribution, deep learning is strongly dependent on massive training data. In common computer vision projects such as self-driving cars and face detection, it is quite easy to acquire large amounts of image and video data. However, for professional communities like disease detection and plant recognition, building a large-scale and high-quality annotated dataset becomes challenging, complex and sometimes expensive [30]. Besides, a large and deep neural network contains a huge number of kernels and weights, which are randomly initialized before training and iteratively updated based on the training data and objective function. Such operation of updating all the weights during training is extremely time consuming. Additionally, with limited training data, deep architectures have the possibility to overfit to the modest dataset. One solution to getting around these problems is to use the *pre-trained* deep learning models for representation feature extraction first, and then to use *transfer learning* to adapt the models to the particular application scenario [31].

Concretely, transfer learning is an important tool in deep learning to solve the basic problem of insufficient training data. It aims at transferring the knowledge from the source domain to the target domain by relaxing the hypothesis that the training data must be independent and identically distributed with the test data. Such rationale provides a promising alternative that makes use of a deep model trained with a common dataset. As discussed above, CNNs are able to learn hierarchical representations from image data, and the knowledge embedded in the kernels/weights of the pre-trained model can be transferred to the new task. Tremendous experiments have shown that, lower-level convolutional layers extract low-level features like edges and curves, which are applicable to common image classification tasks. Operations at higher layers can learn more abstract representations that are specific and relevant to different application fields [24]. Therefore, lower-level representations can be transferred to a new task, and only the higher-level features need to be learned from the new data, even the amount of data is not huge. As such, this will lead to a great positive effect that the deep model in the target domain is not necessary to get trained from scratch, thereby significantly reducing the demand of training data and training time [32], [33]. Such advantages, therefore, motivate us to employ transfer learning in detection

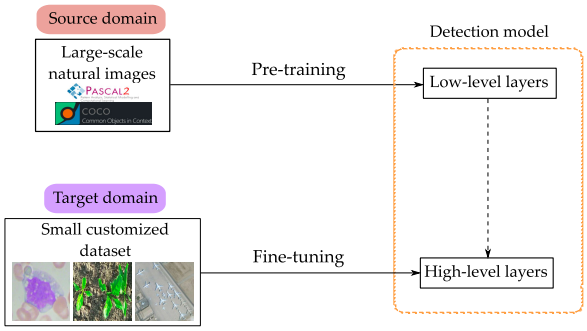


FIGURE 2. Principle of transfer learning. With the help of *transfer learning*, in the *source domain*, large-scale natural images such as Pascal VOC and MS COCO are used to pre-train the low-level layers of the deep detection model. Afterward, learned features are transfered into the *target domain* to equip the model with general image representations. Afterward, small-scale specialized data in professional communities are used to re-train and fine-tune high-level layers.

network training to against problems of insufficient training data and tremendous training time by initializing the target deep model with parameters transferred from a pre-trained model. The basic principle of transfer learning is illustrated in Fig. 2.

The procedure for updating weights of higher layers is called *fine-tuning*. Its success partly relies on the disparity between the source data and the target data. For similar data distribution, one can only fine-tune the fully-connected layers, while for datasets that have considerable differences, several convolutional blocks need to be updated along with training [34]. In this paper, considering the principle of transfer learning and characteristics of the two datasets in two domains, pre-training the deep model shown in Fig. 1 is achieved using large datasets of natural images, Pascal VOC and MS COCO. Concretely, the training scheme is composed of three stages:

1) Train parameters in dashed blocks of the *Darknet-53* module on the MS COCO dataset and freeze the other two modules;
2) Train parameters in dashed blocks of the *Multi-scale* module on the Pascal VOC dataset;
3) Train parameters in dashed blocks of the *Multi-scale* module again and the *Detection* module with respective professional datasets in vision-based projects.

By doing so, weights are updated in respective training stages. As such, model parameters and hyper-parameters can be learned and then transferred to the target domain. Low-level weights are directly obtained from the pre-trained model, and high-level kernels are further fine-tuned for the specific detection tasks. In this way, transfer learning gives the target model a reasonable initialization and reduces the number of parameters that need to be updated, as well as ameliorates the burden of training a large and deep detection model from scratch.

C. IMPROVED ANCHOR BOXES GENERATION

In YOLOv3-based detection model, a representative width and height, or the so-called anchor box, which is a set of predefined bounding box priors, has to be defined *a priori*

to capture the scale and aspect ratio of specific object. They are typically chosen based on object sizes in the training data. Therefore, although YOLOv3 can use any reasonable set of anchor boxes for model convergence, the anchor box can be selected in a targeted manner by analyzing the training samples of the input training dataset, such that we can achieve more effective training convergence. A clustering method, k-means, is employed in YOLOv3 to determine anchor boxes to avoid considerable time-consumption in adjusting the width and height, instead of directly mapping the coordinates of the bounding box. However, there are two theoretical drawbacks of the k-means clustering method. (1) The complexity of the k-means clustering method is expressed as $O(n^{kd})$ for the data based on d dimension and k cluster centers, whereby n is the number of data. The larger the dataset is, the more time-consumption the model processes. (2) The number of clusters k is a user-defined parameter, which means that an inappropriate choice of k may yield poor results. Consequently, the YOLOv3 method is sensitive to the initialization of cluster centers and the anchor boxes found can be thus arbitrarily bad. An inappropriate choice of k may yield poor detection results [35].

To overcome problems that the k-means clustering brings, we propose to apply two optimization strategies: mini batch iteration and heuristic initialization. The first solution is to use the mini batch clustering strategy [36]. The main idea is to randomly use a small and size-fixed batch of samples to reduce the computational burden by not using all samples in each iteration. For the next iteration, a new random batch of samples from the dataset is extracted to update the clusters. As the iteration of clustering goes on, the effect of new mini batch of samples is gradually reduced. Such operation is repeatedly implemented until no changes to the clusters occur in several consecutive iterations and the clustering procedure finishes and converges as a result. Second, to perform the heuristic initialization, k-means++ clustering method is considered here [37], [38]. Compared to randomly specifying initial cluster centers, k-means++ method starts with the allocation of the first cluster center uniformly at random. Afterward, other centers are searched and chosen from the remaining data points with probability proportional to the squared distance from the point's closest existing cluster center given the first one. As such, the improved seeding method ensures a smarter initialization of the centroids and yields considerable improvement in the quality of clustering. A faster convergence and better quality of the final cluster centers is thus guaranteed.

In order to measure the performance of each clustering method, average IoU (shorten as Avg IoU) between boxes that are generated by using cluster centers and all ground-truth boxes is used as a metric of target clustering analysis. The objective function of Avg IoU is written as

$$\text{Avg IoU} = \frac{1}{N}\sum_{j=1}^{N}\max_{i\in[1,\ldots,k]}\left\{\text{IoU}\left(\text{Ground-truth}_i, \text{Prediction}_j\right)\right\}, \quad (1)$$

where N is the number of ground-truth boxes, and k is the number of cluster centers. The larger the Avg IoU value, the better the clustering effect.

III. REAL-TIME DETECTION RESULTS

As is explained above, many vision-based projects require fast object detection. In this section, we examine the proposed method by applying to three typical visual sensing applications, i.e., digital diagnosis (cell detection), smart agriculture (crop detection) and field monitoring (aerial scene detection).

A. ANCHOR BOXES GENERATION AND ACCELERATION

Before proceeding to target detection, we first examine the proposed mini batch k-means++ clustering method and generate anchor boxes for each customized situations. The original YOLOv3, which uses the conventional k-means method for clustering, generates 9 cluster centers (anchor boxes), 3 of which are for each scale by default. According to Ref. [39], for YOLOv3, a higher or lower number of k may increase both the training and validation loss, leading to a worse detection. Therefore, an optimal value is between 6 and 9. After testing the two values, we find that there is no big difference between them, we then follow the routine of the original setting and set k to be 9. Besides, these generated anchor boxes, in a general sense, performs particularly well for the MS COCO dataset. However, for customized datasets in professional communities, it would be better to generate new anchor boxes accordingly. In Table 1, we compare the running time, average IoU and anchor boxes using the conventional and improved k-means clustering methods for the MS COCO data and each detection cases used in this paper. Note that since the MS COCO dataset is only used for image feature extraction, average IoU and newly generated anchor boxes for this case are therefore not given and marked as N/A. As can be seen that the running time for all four cases are reduced due to the involvement of the proposed clustering strategy. Accuracy of average IoU is also improved. This illustrates that the proposed clustering method can lead to a higher average IoU and shorter time consumption. Anchor boxes are accordingly generated for each specialized detection case. In the following experiments, newly generated anchor boxes by the proposed clustering method and the original one are used for detection and comparison.

B. CELL DETECTION

In blood testing, knowing the ratio and throughput of the red blood cell (RBC), white blood cell (WBC) and platelet of a patient is crucial to help doctors make a clinical diagnosis. The first example is to examine the performance in detecting and counting the individual cells. The dataset we use here is Blood Cell Count and Detection (BCCD) dataset, which is a small-scale dataset for blood cells detection.[1] Totally, the dataset contains 364 images and the respective annotations. Each image has a size of 640×480. Annotations are labeled by manually outlining the individual cells and

[1] see https://github.com/Shenggan/BCCD_Dataset

TABLE 1. Comparison of running time, average IoU (%) and generated anchor boxes using conventional and improved *k*-means clustering methods. Best results are marked in bold. The unit of running time is s.

Dataset	Conventional *k*-means Clustering			Proposed Mini Batch *k*-means++ Clustering		
	Time	Avg IoU	Anchor Boxes	Time	Avg IoU	Anchor Boxes
MS COCO	1963.42	N/A	$(10 \times 13), (16 \times 30), (33 \times 23),$ $(30 \times 61), (62 \times 45), (59 \times 119),$ $(116 \times 90), (156 \times 198), (373 \times 326)$	**768.32**	N/A	N/A
Cell	5.36	87.01	$(25 \times 33), (52 \times 91), (59 \times 71),$ $(63 \times 85), (66 \times 101), (72 \times 78),$ $(72 \times 90), (81 \times 100), (140 \times 170)$	**0.68**	**88.03**	$(25 \times 33), (55 \times 90), (61 \times 71),$ $(66 \times 85), (67 \times 102), (72 \times 93),$ $(75 \times 80), (83 \times 102), (140 \times 172)$
Crop	2.11	68.23	$(8 \times 15), (12 \times 11), (15 \times 21),$ $(19 \times 9), (28 \times 15), (33 \times 33),$ $(55 \times 63), (105 \times 108), (166 \times 219)$	**0.59**	**69.69**	$(11 \times 13), (18 \times 10), (18 \times 23),$ $(30 \times 15), (37 \times 35), (63 \times 74),$ $(109 \times 110), (149 \times 202), (245 \times 256)$
Aerial scene	13.68	80.65	$(9 \times 9), (12 \times 13), (17 \times 16),$ $(20 \times 21), (25 \times 25), (31 \times 33),$ $(41 \times 45), (51 \times 59), (147 \times 179)$	**0.81**	**81.11**	$(9 \times 10), (14 \times 14), (19 \times 19),$ $(24 \times 26), (31 \times 33), (41 \times 45),$ $(51 \times 59), (102 \times 174), (172 \times 183)$

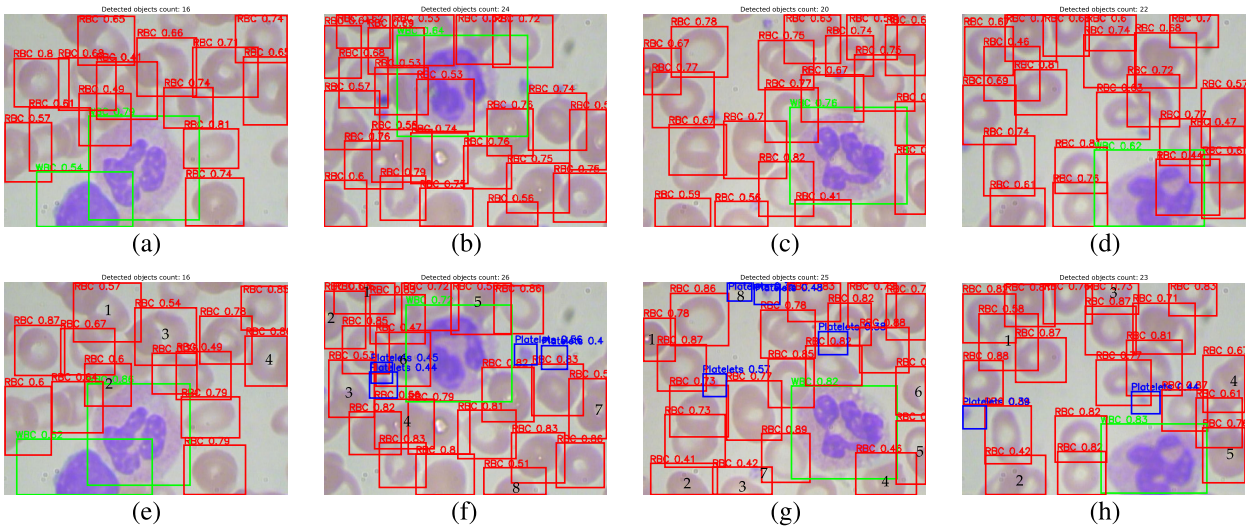


FIGURE 3. Cell detection results of BCCD dataset. (a-d): Using the original detection model. (e-h): Using the proposed detection model.

stored in VOC format. Comparatively, the RBC and WBC are larger and the platelet is smaller, meaning that recognizing the platelet is more challenging. It is also worth to note that these images are captured under a microscope equipped with a high numerical aperture (NA) objective lens, leading to a narrow depth-of-field (DOF). Consequently, when the RBC and WBC are within the DOF and thus in-focus, the platelet may be slightly out-of-focus and blur. This is also true for RBCs and WBCs when multiple cells are located within the field-of-view of the microscope. In some cases, two or more RBCs are so close and clustering that they even have overlapping regions. On the other hand, although RBCs are mostly homogeneous in shape and size, they still vary greatly in morphology and images often contain visible debris. Therefore, we consider these variables more challenging factors and may pose problems for automated detection methods.

For network training, 80% of the data is randomly split for training, 10% is for validation and the remaining 10% is for testing. In Fig. 3, we show four detection and counting results of the individual cell types with their bounding boxes using the original and proposed models. Figures at the top row are created with the classic YOLOv3 model, while figures at the bottom are obtained with the new anchor boxes and transfer learning. Note that cells are marked in red (RBC), green (WBC) and blue (platelet), respectively. On the top of each bounding box, prediction score of each target is also given. From the figures we can see that, most RBCs can be correctly located and recognized by the two methods. However, the proposed method is apparently capable to find more RBCs. Despite several RBCs are severely overlapping, the improved network can still successfully detect them. Even some RBCs that are not annotated in the annotation files can be surprisingly found out. These newly found RBCs are labeled with numbers in Fig. 3. In the four cell images, 4 new RBCs are detected in Fig. 3, while in Fig. 3, as many as 8 new RBCs are noticed. This means that due to the limitation of manually annotated cell images, the authenticity of the annotation files is actually problematic and may lead to an incorrect report of the blood test of a patient, which may negatively affect the clinical diagnosis. We also note that in Fig. 3, the platelet Number 7 is not successfully found, while Number 8 is wrongly recognized as a platelet.

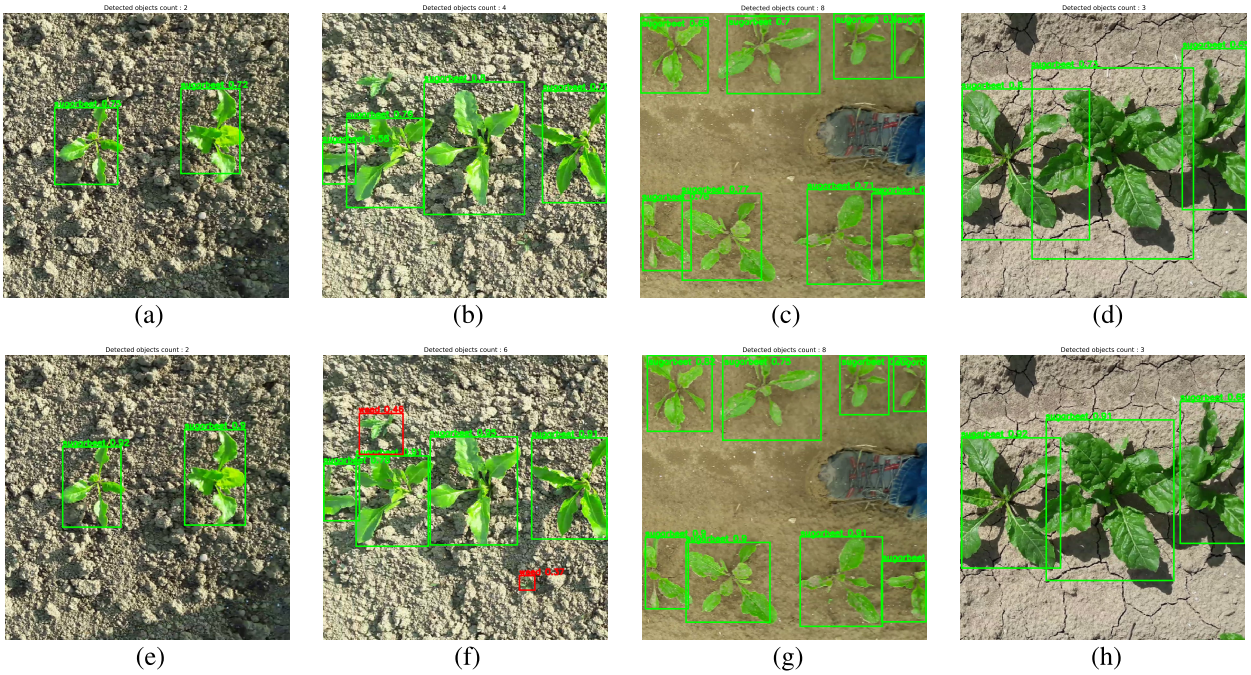


FIGURE 4. Crop detection results. (a-d): Using the original detection model. (e-h): Using the proposed detection model.

Some RBCs close to margins are also not correctly detected. This is because: (1) the cell is not complete and thus the information for a final decision is insufficient; (2) some RBC cells are not annotated in the annotation files, such that in the training stage, the network can barely handle cells locating around borders; (3) in some cases, the platelet is not within the DOF and thus out-of-focus and blur, increasing the difficulty for detection. A solution is to acquire more images to cover as many imaging situations as possible, otherwise one can use defocused images in the pre-training stage to train the network, and then transfer weights to handle such scenario. As for the WBC cell, since there is only one or two WBCs in a single image and the size is large, such that more information and features can be extracted by the network. That's why in Fig. 3, all WBCs can be detected with pretty high scores. For platelets, as mentioned above, in some images they are out-of-focus and blurred. Unclear edges and main structures pose difficulty in detection. However, despite scores are not very high, compared to Figs. 3b-3d obtained with the classic model, the proposed method can still detect them, as shown in Figs. 3f-3h.

C. CROP DETECTION

In smart agriculture, a UAV is frequently deployed to capture images of the crop and to monitor the growth status, soil quality and weed identification. The second example is to achieve crop and weed detection, which is an important topic in agriculture. Concretely, the dataset records two plants: sugarbeet and weed.[2] Totally there are 120 images and annotations. Each image has a size of 512×512. Before training, we artificially augment data by flipping (left-right and up-down) each image and adjusting brightness by multiplying a coefficient to each image. By doing so, the number of images is scaled up by a factor of 3. Then, 80% of the pairs in the new dataset is for training, 10% is for validation and the rest 10% is for testing. In Fig. 4, four candidate images in the testing subset detected using the original and improved methods respectively are shown.

As can be seen, the sugarbeet (marked in green) in a single image is relatively large, compared to the weed (marked in red). Consequently, they are clearly detected by the two methods with relatively high scores, as demonstrated in Figs. 4a, 4c, 4d, 4e, 4g and 4h. Even when only half of the sugarbeet appears in Figs. (4b-4d) and (4-f4h), they can still be successfully found out. However, as shown in Figs. 4b and 4f, only the proposed method can precisely locate the two weeds, one of which is pretty small and hard to be noticed by human eyes. This is the power of the vision-based technologies and the improved model, with which unseen targets can be clearly seen by the machine and algorithm. By taking pictures with a camera set up on a UAV and instantly calculating and transmitting results, people can achieve precise control and management in smart agriculture.

D. AERIAL SCENE DETECTION

Remote sensing image scene detection is an active research topic in the field of aerial and satellite image analysis in the past decades. Due to the involvement of high-resolution, hyper-spectral imaging instruments, it is demanding for intelligent earth observation to develop advanced methods to handle high-performance computing requirements. The third example is to realize land-use detection in remote sensing. The dataset we use here is a collection of aerial scenes

[2] see https://github.com/jmpap/YOLOV2-Tensorflow-2.0/tree/master/data

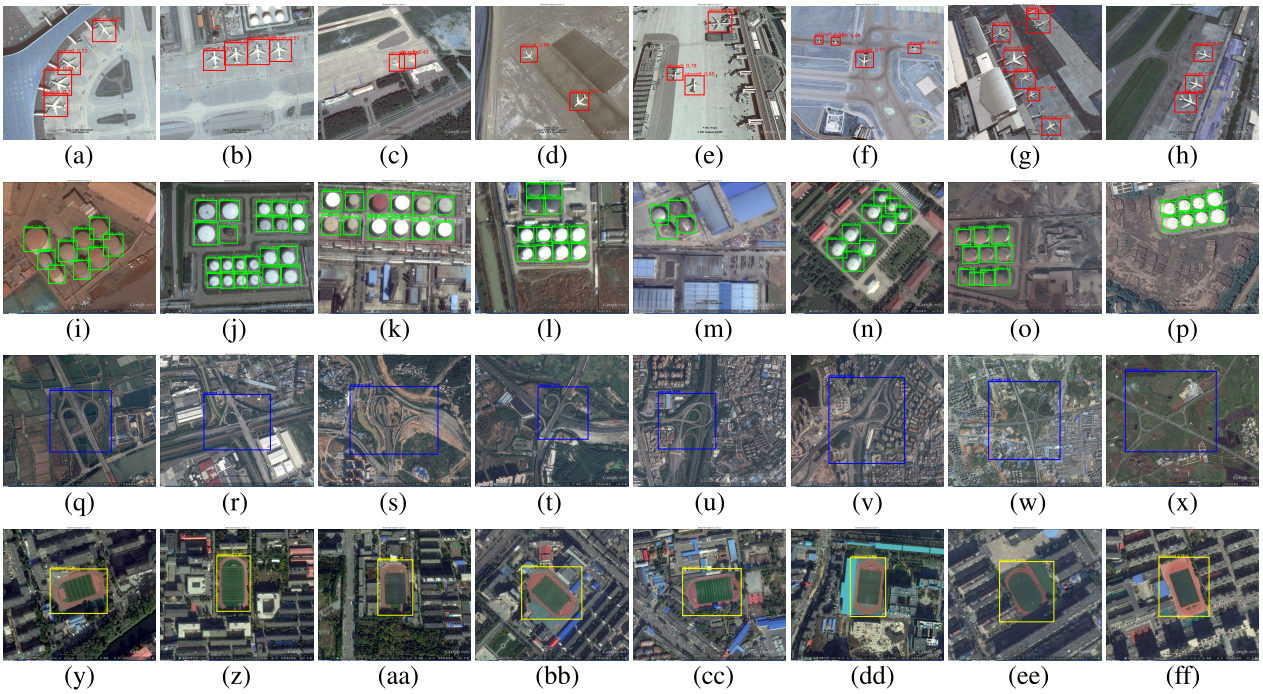


FIGURE 5. Aerial scene detection results of detection results of (a-h) aircraft, (i-p) oil tank, (q-x) overpass and (y-ff) playground in RSOD dataset.

captured from a drone/airborne or space platforms. The Remote Sensing Object Detection (RSOD) dataset[3] includes four scenes, aircraft, oil tank, playground and overpass, collected from Google Earth and Tianditu [40]. There are 446 images for the aircraft, 165 images for the oil tank, 189 images for the playground, and 176 images for the overpass. Each image has a resolution about 1000 × 1000. For each class, 10 images are randomly selected for testing (totally 40 images in the test subset), and the remaining images are for training. In Fig. 5, 8 predicted images selected from each category and created using the improved method are demonstrated (predictions with the conventional method are not shown here since targets can be basically detected but with a relatively lower score), in which the aircraft is labeled in red, oil tank is in green, overpass is in blue and playground is in yellow. The prediction score is also annotated at the top of each bounding box.

As can be seen, these images are captured by an airborne or a satellite with complex backgrounds and surroundings, leading to varying image contrast. For the aircraft and oil tank, there are more than one target of diverse scales and poses in a single image. These aspects raise great challenges in accurate detection. However, the proposed method, as shown in Fig. 5, has a superior performance. Since one image contains only one overpass or playground, it is thus comparatively easier for detection, leading to high confidence scores. For the aircraft and oil tank, all targets are successfully found out, even in cases where targets are closely located. Especially for small aircrafts in Fig. 5f and for oil tanks having a similar color to the background in Figs. 5i and 5o, all targets are successfully identified.

[3]see https://github.com/RSIA-LIESMARS-WHU/RSOD-Dataset-

To further examine the generality of the trained network, we select several remote sensing images from two new datasets, NWPU VHR-10 [41] and NWPU-RESISC45 [42], for detection. The former dataset is a public geospatial object detection dataset, and the latter is a benchmark for remote sensing image scene classification. Each image in NWPU VHR-10 has a size of about 800 × 500, while the image size of NWPU-RESISC45 is 256 × 256. To evaluate the generalization capability of the above trained network, for each class, we select four images from the two datasets, respectively. Then, they are fed into the network for testing, and the detection results are shown in Figs. 6 and 7. It is clearly demonstrated that the trained network is capable to deal with new data from other datasets, thereby confirming that representation features of these four classes, aircraft, oil tank, overpass and playground, are indeed learned by the transfer learning-enabled method. For classes of aircraft, overpass and playground, scores are relatively high. That is because structures and shapes of them does not vary greatly in two datasets. For the oil tank, since the viewing angle and altitude when capturing an image change significantly, the shadow and inclination affect the recognition and therefore scores are not as high as other three categories. However, they are still correctly located and recognized. Consequently, the generality is supported by results in Figs. 6 and 7.

E. PERFORMANCE ANALYSIS

To quantitatively evaluate the proposed method, we conduct comparison experiments on the network with and without transfer learning. Two aspects are examined, performance in object detection and running time for training and testing per image. Here, for the former, we use *recall* and *precision* as objective evaluation metrics. Definitions of

TABLE 2. Performance comparison of models with and without the proposed method. Best results are marked in bold.

Detection Cases	Class	Model with Proposed Method				Model with Original Method			
		Recall	Precision	Training	Inference	Recall	Precision	Training	Inference
Cell	RBC	**0.9582**	**0.9554**	**2.5**	**36.6**	0.8635	0.7521	6.8	37.1
	WBC	**0.9821**	**0.9151**			0.9283	0.8312		
	Platelet	**0.9436**	**0.8988**			0.8369	0.6748		
Crop	Sugarbeet	**0.9721**	**0.9690**	**1.7**	33.2	0.9196	0.9027	5.2	**32.9**
	Weed	**0.9328**	**0.9197**			0.7655	0.5604		
Aerial Scene	Aircraft	**0.9428**	**0.9371**	**3.3**	**38.4**	0.8771	0.8222	10.5	38.5
	Oil tank	**0.9458**	**0.9588**			0.8747	0.8399		
	Overpass	**0.9244**	**0.9051**			0.8933	0.8674		
	Playground	**0.9742**	**0.9431**			0.9384	0.9188		

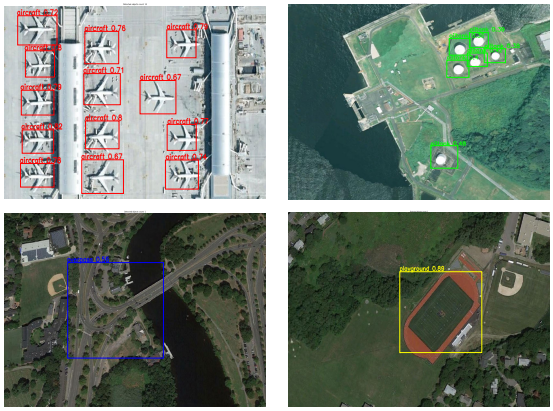


FIGURE 6. Detection results of aircraft, oil tank, overpass and playground in NWPU VHR-10 dataset.

FIGURE 7. Detection results of aircraft, oil tank, overpass and playground in NWPU-RESISC45 dataset.

recall and precision are, respectively, the ratio of correctly detected objects to the total number of actual objects, and the ratio of correctly detected objects to all detections in the actual classes. Scores and time consumptions of training and inference are given in Table 2. The unit of training time is h, and the unit of inference time is ms. Results indicate that, with the help of transfer learning, the model performs better for all categories in three cases, as the bold scores indicate. Despite the performance of the improved YOLOv3, compared to other detection frameworks, is not state-of-the-art, refinements with the help of the advanced clustering method and deep transfer learning are demonstrated, which is the novelty and key point of this paper. Furthermore, although the inference time per image with and without transferring pre-trained weights is basically identical since the proposed clustering method reduces the complexity of anchor boxes generation, which is implemented outside the main detection architecture of YOLOv3, training time is significantly reduced by 4 h("Cell" and "Crop") or 6 h("Aerial Scene"), saving plenty of time in potentially configuring the model to new scenarios. Therefore, the engagement of transfer learning can indeed boost the network performance, and the burden of training a model from scratch is greatly ameliorated.

In order to test whether the proposed method performs better than the original approach with the statistical significance, we implement the *paired t-test* under the three cases. The paired t-test is a common way to test whether the difference between two measurements over various data sets is non-random. [43]. Let d_i be the difference between the performance scores of the two detectors on the i-th out of N scores of recall and precision ($N = 18$ as shown in Table 2). The mean difference $\overline{d}$, standard deviation of the differences σ_d, standard error of the mean difference $\mathrm{SE}(\overline{d})$, and the t-statistic T_{paired} are thus computed as

$$\overline{d} = \frac{1}{N}\sum_{i=1}^{N} d_i = 0.1062, \tag{2}$$

$$\sigma_d = \sqrt{\frac{\sum_{i=1}^{N}(d_i - \overline{d})^2}{N-1}} = 0.0854, \tag{3}$$

$$\mathrm{SE}(\overline{d}) = \frac{\sigma_d}{\sqrt{N}} = 0.0201, \tag{4}$$

$$T_{\text{paired}} = \frac{\overline{d}}{SE(\overline{d})} = 5.2736. \tag{5}$$

According to the t-distribution with $\mathrm{df} = N - 1 = 17$ degrees of freedom, with a specified alpha level of 0.05 (5%), from the t-table we have $T_{(\mathrm{df}=17,\alpha=0.01)} = 2.110$. Since

TABLE 3. Ablation study of contributions from the clustering method and the pre-training method. The unit of training time is h, and the unit of inference time is ms.

Class	Model with Clustering Only				Model with Pre-training Only			
	Recall	Precision	Training	Inference	Recall	Precision	Training	Inference
Aircraft	0.9012	0.8978	10.3	37.8	0.8843	0.8752	3.6	39.2
Oil tank	0.9159	0.9197			0.8923	0.9038		
Overpass	0.9070	0.9022			0.9008	0.8875		
Playground	0.9568	0.9410			0.9259	0.9301		

$T_{\text{paired}} > T_{(\text{df}=17,\alpha=0.01)}$, we can then reject the null hypothesis that there is no significant difference between the two approaches. Instead, the significance testing shows strong evidence that, on average, the proposed module does lead to detection improvements compared to the original approach.

Furthermore, to evaluate the influence of performance enhancement provided by the advanced clustering and the pre-training, we conduct ablation study by remaining each method, respectively. Table 3 presents the experimental result that is performed with the case of remote sensing. As can be seen, the detection models under both cases, clustering only and pre-training only, perform better than the original model and worse than the proposed method in recall and precision. This is reasonable since the advanced clustering, which provides better anchor boxes, and the pre-training strategy, which extracts comprehensive features from more images, have a positive impact on the detection model. A minor improvement is thus achieved in Table 3. As for the running time, the scheme of pre-training accelarates the training stage as the proposed method, while the model only with clustering has a similar performance to the original method. While the inference time basically keeps the same to scores in Table 2, since the network architecture does not change and the feedforward computation remains. All in all, from the ablation study we can conclude that, separate involvement of the clustering method and the pre-training scheme is solely capable of providing a limited improvement in detection. By combining the two strategies together, a far better performance in detection can be successfully achieved.

IV. CONCLUSION

In this paper, we have developed a novel deep transfer learning framework and improved detection model for real-time object detection in vision-enabled environments. Main contributions lie in three aspects as follows.

- By using the mini batch k-means++ clustering method, computational burden of generating new anchor boxes for customized data is significantly reduced and the quality of new cluster centers is improved.
- By pre-training the detection model with natural images and by transferring weights to customized datasets, time-consuming network training from scratch is thus avoided and superior detection performance is achieved even with insufficient data.
- Comparative analysis shows that the method can handle multiple targets detection at about 30 FPS (i.e., 33.2 ms)@416×416 per image across each of these datasets with pretty high confidence scores, in a general sense.

We validate the proposed method and demonstrate its outstanding performance by applying to three typical vision-based sensing applications, disease diagnosis, smart agriculture and earth observation. With the help of powerful and advanced learning algorithms, we envision that in the future they can be expected to play an essential and constructive role in object detection and other processing tasks across diverse vision-based scenarios.

ACKNOWLEDGMENT

Zhenbo Ren thanks Dr. N. Meng from The University of Hong Kong for fruitful discussions.

REFERENCES

[1] H. Li, P. Wang, and C. Shen, "Toward end-to-end car license plate detection and recognition with deep neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 1126–1136, Mar. 2019.

[2] D. S. W. Ting, Y. Liu, P. Burlina, X. Xu, N. M. Bressler, and T. Y. Wong, "AI for medical imaging goes deep," *Nature Med.*, vol. 24, no. 5, pp. 539–540, May 2018.

[3] R. Mlambo, I. Woodhouse, F. Gerard, and K. Anderson, "Structure from motion (SfM) photogrammetry with drone data: A low cost method for monitoring greenhouse gas emissions from forests in developing countries," *Forests*, vol. 8, no. 3, p. 68, Mar. 2017.

[4] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. T. Campbell, "A survey of mobile phone sensing," *IEEE Commun. Mag.*, vol. 48, no. 9, pp. 140–150, Sep. 2010.

[5] R. Crane, "Automatic cell detection and tracking," *IEEE Trans. Geosci. Electron.*, vol. 17, no. 4, pp. 250–262, Oct. 1979.

[6] A. Kamilaris and F. X. Prenafeta-Boldú, "A review of the use of convolutional neural networks in agriculture," *J. Agricult. Sci.*, vol. 156, no. 3, pp. 312–322, Apr. 2018.

[7] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, opportunities, and directions," *IEEE Trans. Ind. Informat.*, vol. 14, no. 11, pp. 4724–4734, Nov. 2018.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[9] Z. Ren, Z. Xu, and E. Y. Lam, "End-to-end deep learning framework for digital holographic reconstruction," *Adv. Photon.*, vol. 1, no. 1, 2019, Art. no. 016004.

[10] Z. Ren, H. K.-H. So, and E. Y. Lam, "Fringe pattern improvement and super-resolution using deep learning in digital holography," *IEEE Trans. Ind. Informat.*, vol. 15, no. 11, pp. 6179–6186, Nov. 2019.

[11] D. S. W. Ting, C. Y.-L. Cheung, G. Lim, G. S. W. Tan, N. D. Quang, A. Gan, H. Hamzah, R. Garcia-Franco, I. Y. S. Yeo, S. Y. Lee, and E. Y. M. Wong, "Development and validation of a deep learning system for diabetic retinopathy and related eye diseases using retinal images from multiethnic populations with diabetes," *JAMA*, vol. 318, no. 22, p. 2211, Dec. 2017.

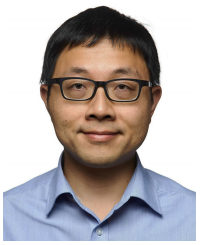
[12] K. Sirinukunwattana, S. E. A. Raza, Y.-W. Tsang, D. R. J. Snead, I. A. Cree, and N. M. Rajpoot, "Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1196–1206, May 2016.

[13] L. Shen, L. R. Margolies, J. H. Rothstein, E. Fluder, R. McBride, and W. Sieh, "Deep learning to improve breast cancer detection on screening mammography," *Sci. Rep.*, vol. 9, no. 1, pp. 1–12, Dec. 2019.

[14] S. Sarraf and G. Tofighi, "Classification of Alzheimer's disease using fMRI data and deep learning convolutional neural networks," 2016, *arXiv:1603.08631*. [Online]. Available: http://arxiv.org/abs/1603.08631
[15] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 781–788, Apr. 2017.
[16] M. Bah, A. Hafiane, and R. Canals, "Deep learning with unsupervised data labeling for weed detection in line crops in UAV images," *Remote Sens.*, vol. 10, no. 11, p. 1690, Oct. 2018.
[17] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016.
[18] S. H. Lee, C. S. Chan, P. Wilkin, and P. Remagnino, "Deep-plant: Plant identification with convolutional neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 452–456.
[19] M. Mahdianpari, B. Salehi, M. Rezaee, F. Mohammadimanesh, and Y. Zhang, "Very deep convolutional neural networks for complex land cover mapping using multispectral remote sensing imagery," *Remote Sens.*, vol. 10, no. 7, p. 1119, Jul. 2018.
[20] E. Ndikumana, D. Ho Tong Minh, N. Baghdadi, D. Courault, and L. Hossard, "Deep recurrent neural network for agricultural classification using multitemporal SAR Sentinel-1 for Camargue, France," *Remote Sens.*, vol. 10, no. 8, p. 1217, Aug. 2018.
[21] F. Chen, R. Ren, T. Van De Voorde, W. Xu, G. Zhou, and Y. Zhou, "Fast automatic airport detection in remote sensing images using convolutional neural networks," *Remote Sens.*, vol. 10, no. 3, p. 443, Mar. 2018.
[22] T. R. Martha, N. Kerle, C. J. van Westen, V. Jetten, and K. V. Kumar, "Segment optimization and data-driven thresholding for knowledge-based landslide detection by object-based image analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 12, pp. 4928–4943, Dec. 2011.
[23] S. Ji, C. Zhang, A. Xu, Y. Shi, and Y. Duan, "3D convolutional neural networks for crop classification with multi-temporal remote sensing images," *Remote Sens.*, vol. 10, no. 2, p. 75, Jan. 2018.
[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org
[25] L. Jiao, F. Zhang, F. Liu, S. Yang, L. Li, Z. Feng, and R. Qu, "A survey of deep learning-based object detection," *IEEE Access*, vol. 7, pp. 128837–128868, 2019.
[26] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *Int. J. Comput. Vis.*, vol. 128, no. 2, pp. 261–318, Feb. 2020.
[27] 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.
[28] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767
[29] Ju, Luo, Wang, Hui, and Chang, "The application of improved YOLO v3 in multi-scale target detection," *Appl. Sci.*, vol. 9, no. 18, p. 3775, Sep. 2019.
[30] I. Athanasiadis, P. Mousouliotis, and L. Petrou, "A framework of transfer learning in object detection for embedded systems," 2018, *arXiv:1811.04863*. [Online]. Available: http://arxiv.org/abs/1811.04863
[31] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.
[32] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. Neural Netw.*, 2018, pp. 270–279.
[33] S. Jialin Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[34] O. Day and T. M. Khoshgoftaar, "A survey on heterogeneous transfer learning," *J. Big Data*, vol. 4, no. 1, p. 29, Dec. 2017.
[35] L. Zhao and S. Li, "Object detection algorithm based on improved YOLOv3," *Electronics*, vol. 9, no. 3, p. 537, Mar. 2020.
[36] D. Sculley, "Web-scale k-means clustering," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, 2010, pp. 1177–1178.
[37] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding," in *Proc. 18th Annu. ACM-SIAM Symp. Discrete Algorithms*, no. 9. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.
[38] J. Li, J. Gu, Z. Huang, and J. Wen, "Application research of improved YOLO v3 algorithm in PCB electronic component detection," *Appl. Sci.*, vol. 9, no. 18, p. 3750, Sep. 2019.
[39] P. Hurtik, V. Molek, J. Hula, M. Vajgl, P. Vlasanek, and T. Nejezchleba, "Poly-YOLO: Higher speed, more precise detection and instance segmentation for YOLOv3," 2020, *arXiv:2005.13243*. [Online]. Available: http://arxiv.org/abs/2005.13243
[40] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2486–2498, May 2017.
[41] G. Cheng, P. Zhou, and J. Han, "Learning rotation-invariant convolutional neural networks for object detection in VHR optical remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 12, pp. 7405–7415, Dec. 2016.
[42] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.
[43] N. Meng, X. Sun, H. K.-H. So, and E. Y. Lam, "Computational light field generation using deep nonparametric Bayesian learning," *IEEE Access*, vol. 7, pp. 24990–25000, 2019.

ZHENBO REN received the B.S. degree in optoelectronic information engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2011, the M.S. degree in instrument science and technology from Tsinghua University, Beijing, China, in 2014, and the Ph.D. degree in optical engineering from The University of Hong Kong, Hong Kong, in 2018.

He is currently an Assistant Professor with the School of Physical Science and Technology, Northwestern Polytechnical University, Xi'an, China. His research interests include digital holography, optical imaging, and deep learning.



EDMUND Y. LAM (Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, USA, in 1995, 1996, and 2000, respectively.

He is currently a Professor in electrical and electronic engineering, an Associate Dean of engineering, the Director of the Computer Engineering Program, and the Founding Director with the Imaging Systems Laboratory, The University of Hong Kong, Hong Kong. He has authored or coauthored over 300 journal and conference papers. His research interests include computational optics and imaging.

Dr. Lam was a Fellow of the Society of Photo-Optical Instrumentation Engineers (SPIE), OSA, the Society for Imaging Science and Technology, and the Hong Kong Institution of Engineers. He was a recipient of the IBM Faculty Award. He serves as the Chair for the Optical Society (OSA) Image Sensing and Pattern Recognition Technical Group and the Chair for the Computational Optical Sensing and Imaging Meeting in 2019.

JIANLIN ZHAO received the M.S. degree from Northwestern Polytechnical University, in 1987, and the Ph.D. degree from the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, in 1998.

He is currently a Professor with the School of Physical Science and Technology, Northwestern Polytechnical University, China. He is also the Director of the MOE Key Laboratory of Material Physics and Chemistry under Extraordinary Conditions and the Shaanxi Key Laboratory of Optical Information Technology. His research interests include digital holography, micro-nano photonics, and optical fiber sensor.