International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Aircraft Detection System Based on Regions with Convolutional Neural Networks

Ferhat Ucar*[1], Besir Dandil[2], Fikret Ata[3]

Submitted: 21/02/2020 Accepted : 07/08/2020

Abstract: Object detection in remote sensing imagery is an important topic of image processing researches. Detection of objects and regions from satellite imagery takes its place in various applications such as the detection of residential areas and agricultural lands, road lines, ships, as well as airport and hangar detections. As a more specific remote sensing imagery based object-detection process, a stationary aircraft detection system could serve as a model in some military applications. Such a model could serve the detection of stationary aircraft targets in airports. In the proposed study, a deep learning-based model detects the aircraft in the airports using the satellite images from Google Earth. The deep learning model uses the state of the art Regions with Convolutional Neural Network (RCNN). Firstly a built from scratch CNN design is used for the basic learning step of the system. Then, RCNN performs region detection, which anchors the stationary aircraft object bounding boxes. A large dataset containing aircraft images is preferred for training of CNN. To validate the system, satellite images captured from airports in Turkey are used. The results of the study show that the proposed model successfully operates aircraft detection with high-performance rates. While the classifier network structure, which constitutes the first step of the study, produces 98.4% test accuracy, the proposed aircraft detection framework has successfully performed the aircraft identification process by producing matched bounding boxes in the test images.

Keywords: *Aircraft Detection, Deep Learning, Regions with Convolutional Neural Networks, Remote Sensing*

1. Introduction

Today, object detection and identification in remote sensing images are among the prominent topics in satellite image processing. The freedom to obtain high-quality images from a desired region of the earth, which is especially revealed by the developments in satellite technologies, leads to the diversification and development of the studies in this field day by day. Topics studied in the field of satellite images can range from storm density determinations that support climatic forecasts to the identification of agricultural crops and income determination of residential areas. The processing and understanding of remote sensing images also play an important role, especially in a number of military applications and in many practices that fall within the scope of the defence industry. In the target detection process used in applications that form the backbone of defence systems such as unmanned aerial vehicles, important outputs can be presented to decision support systems with the help of high-level information that can be obtained from the images.[1], [2].

In general, if it is desired to define the problem of object recognition from high-resolution images, it can be said that this definition is based on determining the position of the object in the given image. Another component of this definition is to perform the classification of the detected image. Conventional object recognition systems designed on these two components include the phases as 1) the identification of regions containing information, 2) feature extraction and 3) classification [1], [3], [4]. In applications that rely on basic image processing methods, the importance of the feature extraction step is undeniable. In cases where the number of samples in the data set increases, the feature extraction process performed with the transformation-based basic methods can be complicated in terms of the computational cost and the possible effect of the outputs on classifier generalization performance [5]. The process, which started with the development of deep learning methods by researchers, makes important contributions to the feature extraction step, which is the most critical component of the system. Because of its nature, the feature extraction process in the deep learning model becomes automatically mapped within the algorithm flow, different from the handcrafted structure in conventional applications. In recent years, research studies of object detection applications based on deep learning methods have been increasing rapidly with the overall development of computer hardware technologies and especially with the improvements in graphical card memory capacities [6], [7].

Especially the necessity of "region-based" workflow in the object recognition process to be applied to remote sensing images has caused the need to reflect this development to deep learning models. In the model proposed by Girshick et al. [8] as part of a competition held in 2012, the Regions with Convolutional Neural Network (RCNN) structure has bridged the gap between image classification and object recognition processes. In this method, a feature extraction process based on the CNN structure and the resulting region proposals are classified with a Support Vector Machine (SVM). Thus, the region with the highest classification result is determined by assigning it to the class it is included. The

[1] *Electrical and Electronics Eng. Firat University, Elazig – 23019, Turkey ORCID ID: 0000-0001-9366-6124*
[2] *Mechatronics Eng. Firat University, Elazig – 23019, Turkey ORCID ID: 0000-0002-3625-5027*
[3] *Electrical and Electronics Eng. Bingol University, Bingol – 12100, Turkey ORCID ID: 0000-0003-1100-6179*

** Corresponding Author Email: fucar@firat.edu.tr*

object is then indicated by a bounding box [8], [9]. Throughout the developing process, deep learning models are proposed, which perform object recognition in many different structures created by researchers with RCNN [10]–[12]. An application design, which carries out vehicle detection within the scope of traffic surveillance with RCNN, is presented with the study of [13]. An application in which the determination of the eddy events occurring in the oceans with the RCNN structure is also presented in the study of [14]. The effective performance of the RCNN structure in object recognition has allowed the method to become widespread in many different areas.

Aircraft detection and identification systems using remote sensing images have an important place in the object recognition literature, both as conventional image processing methods are used and deep learning-based models are recommended [15]–[18]. In the study [19], as conventional methods, the Gabor filter is used in the feature extraction process and the aircraft detection process is carried out by image classification with the SVM classifier. Conventional methods lag behind in producing effective outputs in the bounding box display where the object is detected in a large image [15]. In studies where basic CNN forms are applied, the searching and detection algorithms performed on the satellite image for the position of the bounding boxes after the classification process also bring an additional computational cost [20]. For this reason, researchers have also proposed multiple formations integrated with basic image processing methods [21]. The transfer learning method, which is preferred in the presence of large data sets and uses pre-trained network structures with various classes, is also used in aircraft detection researches [22]. The prominent disadvantage of this method is the additional computational cost of not using an application-specific deep learning network structure [5]. Besides RCNN, aircraft detection systems using the You Only Look Once (YOLO) structure, which is highly preferred in object recognition applications, have also been introduced in the literature [16].

In this study, an RCNN deep learning model designed to perform the detection of stationary aircraft objects in satellite images is proposed. The proposed model uses a CNN network built from scratch to create design simplicity and much less computational cost. A data set containing small-sized aircraft images are used for the proposed model to perform aircraft detection from scale-invariant satellite images [23]. Using this large data set containing “Plane” and “NPlane” class labels, CNN training is carried out with a variety of images with and without aircraft. The CNN structure designed specifically for the application without using the transfer learning model has made an important contribution in terms of both computational cost and performance. The CNN model used in the first step of our study can produce a test performance value of 98.4% in the classification process. Unlike its initial structure, the RCNN model designed in the study uses the "softmax" layer, which is the natural classifier of a deep network, instead of using SVM in the classification process. This difference can be noted as another contribution to the proposed study. In order to carry out a holistic evaluation of the proposed model, a special dataset consisting of satellite images obtained with the help of Google Earth was created during the RCNN training phase and bounding box labeling was done. In the verification process of the RCNN based aircraft detection model, an additional satellite image dataset, which was not included in the training was collected and labeled. The additional datasets, which are specific for our study, were obtained from airports all around Turkey. The test results show that the proposed model effectively detects aircraft objects and can produce bounding boxes. The contributions of our study can be summarized as follows:

1. The CNN network, which forms the backbone of the proposed RCNN aircraft detection model, is built from scratch. In other words, the network structure created specifically for the application without the transfer learning method is preferred, and it is designed more flexibly away from the strict form of the pre-trained networks.
2. RCNN aircraft detection model stands out with its low computational cost contribution, based on the main backbone CNN structure, trained and tested with a wide range of data, unlike the basic RCNN method, without using SVM. Thus, when tested with satellite images created independently from the scale, the proposed model can effectively perform aircraft detection.

The remaining of the text is organized as follows: Chapter 2 presents the outline of the method of the proposed aircraft detection system with a CNN explanation in brief. Chapter 3 provides the details of the methodology of the proposed model by presenting a description of the datasets used in the study and the experimental studies performed. Findings from the experiments are also presented in this section. In chapter 4, the conclusion paragraph is included.

2. Methodology

In this section, the method details of the proposed deep learning-based aircraft detection system are given. It is intended to draw a framework that addresses the entire system and outlines of the RCNN aircraft detection model using the CNN backbone. The CNN structure, which forms the backbone of the aircraft detection system proposed in the study, is a basic deep learning algorithm. Since the model used is intended to be simple and to increase its applicability, a CNN network specific to this application has been created to form the infrastructure of the aircraft detection application instead of starting with pre-trained networks. The design phases of the method outlined in Figure 1 can be listed as follows:

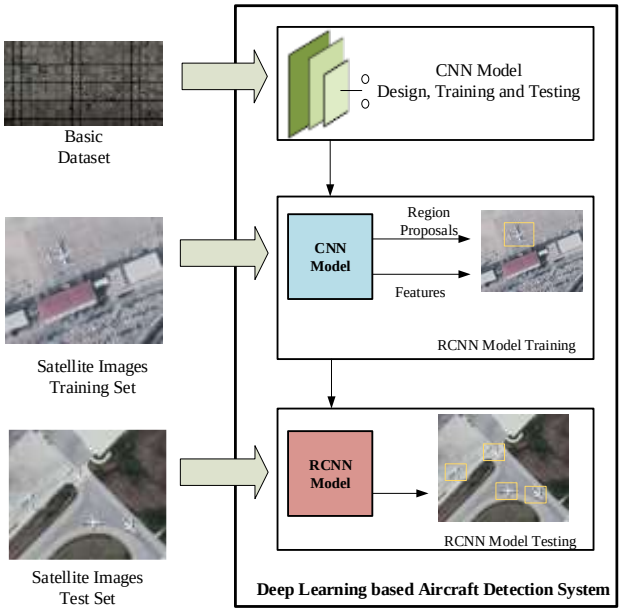

Fig. 1. Block diagram showing the outline of the proposed model

1. Design of CNN model and training and testing with "Plane - Nplane" binary class-based big data set.

2. Training the RCNN model with large satellite images so that it can act as an "object detection" module based on the CNN model, which has been designed and completed in advance. Subsequently, testing with satellite images created by random selection.

2.1. Deep Learning Background of The Model

Deep learning is a subclass of machine learning. The widespread use of this learning structure in many areas as it is today is linked to the development of CNN architectures [5], [24]. CNN is a network structure built on the basis of multilayer perceptron architecture. In the evolutionary learning structure, which is based on the biological vision process, the vision detectors covering the entire image are divided into sub-regions. Simple cell structures in the lower regions continue to focus on the edge and similar features scanned in the image, while complex cell structures continue to focus on larger receivers that scan the entire image. This process defines the convolutional mathematical operation as the basic concept. If mathematical convolution is considered by adapting it to the basis of biological vision, it can be described as the neuron's response to stimuli in its field [5], [6], [25].

The first phase of the CNN structure begins with the "convolution" process. "Pooling" is performed after every convolution process. According to the depth of the designed network, the convolution and pooling layers continue one after another, revealing the deep learning architecture. The CNN architecture ends with "fully connected" layers. At the output, the "softmax" layer generally assists the classification layer. CNN training is continued by using the backpropagation of errors. Thus, as a result of each iteration, all weights in the layers are updated and the error value is minimized. CNN structures, which are generally used in image processing applications, have also been applied with one-dimensional structures for use in sound processing and time series prediction and classification processes with the widespread use of their fields in recent years [5], [6], [26], [27].

In figure 2, the schematic representation of the general layers in the CNN structure is presented. Brief explanations of the layers mentioned here is explained below.

Convolution Layer: Here, the X_i^{l-1} parameter, which expresses the feature map of the previous layer, is transformed with the kernel k_{ij}^l, and the result of the activation function $f(\cdot)$ is obtained by adding b_j, a trainable bias parameter.

Thus, the output feature map is expressed as follows [12]:

$$X_j^l = f(\sum_{i \in M_j} X_i^{l-1} * k_{ij}^l + b_j^l) \quad (1)$$

Here, M_j shows an array selected from the input feature maps. In this study, Rectified Linear Unit (ReLU) is preferred as activation function.

Pooling Layer: In general, this layer is known as the subsampling layer. It transmits the input feature maps to its output with reduced sample numbers. Considering the meaning of image features, this means that the number of input and output features remains the same, but the output features are smaller in size [5].

Fully Connected (FC) Layer: The features from many convolution layers and pooling layers are processed here with a conventional neural network structure. The artificial neural network neurons here are linked to all weights in the previous layer [25].

2.2. RCNN based Aircraft Detector

The second phase of the proposed deep learning-based aircraft detection system involves the design of the RCNN structure of the detector built on the CNN backbone trained with aircraft images. Here, the RCNN structure performs some kind of retraining, using the CNN structure, which was trained and tested with aircraft and non-aircraft images in the previous step [8]. With the new data set obtained in high resolution and scale-invariant, a feature map of region proposals is obtained from the pre-trained CNN structure, and aircraft detection is carried out by locating a bounding box according to the classification result of these regions.

In the object classification process, the class of the input images and the probability that the input belongs to the classes are reflected in the output as a percentage. When it comes to object detection, the situation changes and gets more complicated. Because, in addition to the classes and probability values of the object or objects at the output, the bounding box placement is performed, which gives position determination. For this purpose, possible coordinate information of the object is also given at the output. With the RCNN architecture, which was first introduced by Girshick and his research group, an important performance improvement step was obtained in object detection applications [8].

The first of the RCNN implementation steps is the process of obtaining region proposals. For this purpose, regions are obtained by scanning the input image with the selective search algorithm detailed in the study [28]. In the second step, these region proposals are subjected to feature mapping by the CNN structure before classification. In the next step, SVM training is carried out for the classification process. Here, the output of the candidate regions belonging to which object or object classes is reached. In the last operation, using the linear regression, bounding box placement is made according to the coordinate data obtained from the result values. Thus, the RCNN structure gains a detector structure designed according to the trained object class [8], [29].

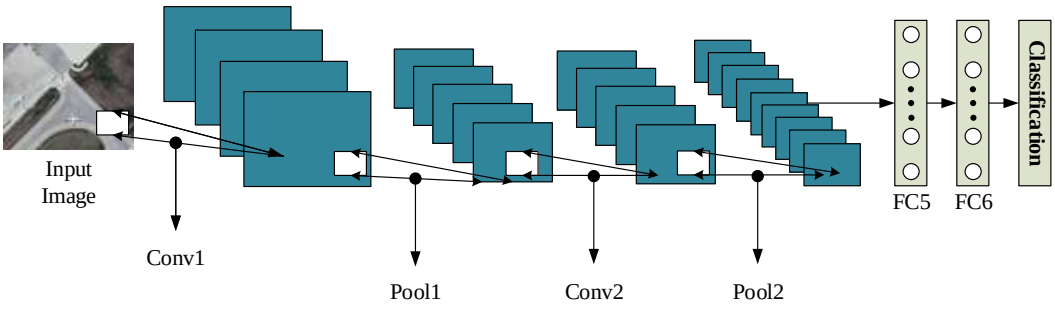

Fig. 2. General flowchart of a CNN network

In the proposed study, unlike the basic RCNN, besides the CNN architecture designed specifically from scratch, the classification was carried out with the help of the softmax layer, without preferring the SVM in the classification. This process also avoids the computational complexity considering the holistic structure of the application. For more information on RCNN, studies [8], [10] can be consulted.

3. Model Setup and Findings

In this section, after giving brief information about the datasets used in the study, the details of the experimental works are mentioned. In addition, the findings obtained as a result of the experiments are presented in the form of tables and images in the last part of the section.

3.1. The Aircraft Dataset

Two different types of datasets are used in the study. The first is the large dataset consisting of smaller images, in which the CNN structure, which forms the backbone of the aircraft detection model, is trained, and the second is the dataset obtained by the authors via Google Earth. The latter one is consisting of large-scale satellite images but with a fewer number of images.
The aircraft dataset used for CNN training and testing is publicly available [23]. Table 1 contains summary information about the dataset.

Table 1. Dataset information that is used with the CNN

Class Label	Number of Images	Image Size / Type
Plane	8,000	20×20 / RGB
NPlane	24,000	

As can be seen from Table 1, the total number of images in the dataset is 32,000. Eight thousand images in this aircraft dataset prepared from satellite images in 2017 belong to the “Plane” class. Images in this class are cropped in approximately the center of the body of a single aircraft body. Examples of different aircraft types are diversified with images containing different layouts and various structures of atmospheric conditions. Aircraft images belonging to this class have been determined to include a large part of aircraft wings, tail parts, and nose sides.
The other class in the data set includes images in the category "not aircraft" (NPlane). There are 24,000 images in this class. One-third of the total class consists of images containing different land features, including sea, vegetation, soil-covered areas, buildings, etc. The other third is composed of images that contain only a small part of the plane, but also do not carry the characteristics of the “Plane” class. The last third part consists of images containing bright objects in the form that can be called "disturbance inputs" or sharp linear features that can remind an aircraft. Some examples of images randomly selected from the dataset are shown with the class tags in Figure 3. For more detailed information about the dataset and downloadable files, the Kaggle machine learning datasets sharing site [23] can be reached.

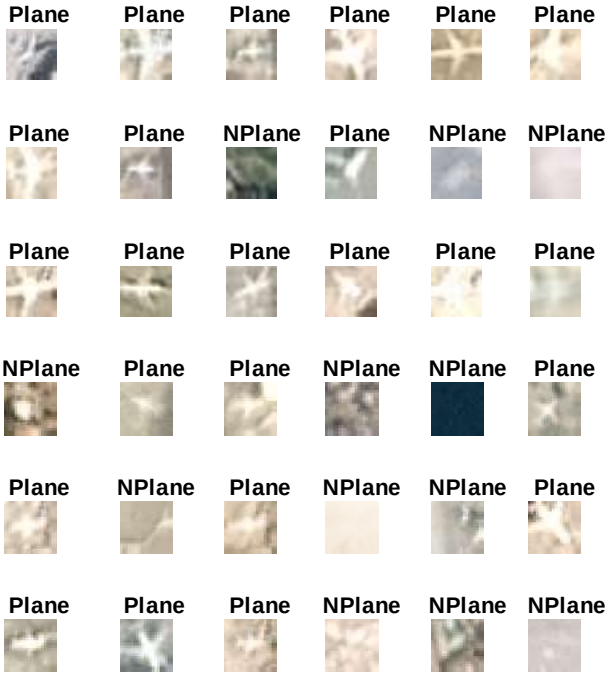

Fig. 3. Sample images of aircraft dataset

In the second step of the study, a unique dataset is created by the authors, using Google Earth for training and testing of the RCNN aircraft detection model. In order to realize the training and test stages as robust form, images independent of scale and elevation are preferred. The dataset is collected from randomly selected airports across Turkey. Images contain stationary aircraft objects with single or multiple distributions. In Figure 4, sample images selected from the satellite images data set created for RCNN training are given. In RCNN training, the bounding boxes must also be placed in order to label the images in the data set and determine their coordinates.

Fig. 4. Sample images selected from the dataset used in RCNN training

In this study, aircraft labels in the images that compose the RCNN training dataset are created manually for each image using the "MATLAB Image Labeler" interface. As the basic part of the study is the dataset in CNN training, we do not need lots of images in the RCNN training dataset. Models with high-test performance can be designed without the need for a large number of samples in the dataset. Therefore, 40 different random satellite images are used in our training dataset. Class information and bounding box coordinate information (annotations) of these images are also created as a table.
Tiny-cropped images containing stationary aircraft forms in an airport satellite image anywhere in the world can be used to create the dataset to be practiced in the RCNN testing phase. In this study, the test dataset is collected from Turkey airports. Sample images

randomly selected from the test dataset, which is including 20 images, are given in Figure 5. Detailed information on the use of "MATLAB Image Labeler" can be accessed with [30].

Fig. 5. Sample images selected from the dataset used in RCNN testing

3.2. Experimental Case

The dataset used to perform CNN training does not have a balanced distribution between classes as detailed in Section 3.1. The effect of the dominant class in the weight renewal process of unbalanced distribution datasets reflects negative effects on the performance [31]. For this reason, the dataset has been reduced to a total of 16,000 images, an equal number of "Plane" and "NPlane" classes. While making the reduction process, the selection was made with random indices from the whole dataset. Training and test rates were determined as 0.7 and 0.3, respectively. Thus, 11,200 images were used in the training set and 4,800 images were used in the test set. RCNN training and test datasets consist of 40 labeled and 20 unlabeled images, respectively.

Within the scope of the study, essentially the CNN architecture that forms the backbone structure was designed. The layers that make up the CNN network built from scratch and application-specific without using the transfer learning method can be summarized as follows: In the architecture that starts with the input layer, the following layer is the convolution (Conv.) layer with 32 channel convolution filters. The convolution outputs are passed through the ReLu activation function. The pooling process was also applied at this layer output and a window size of 2×2 was preferred as the maximum sampling operator. These values are taken into consideration since the images in the data set are 20×20 in size. The next two layers are convolutional layers, and the first has 64 channel convolutional structures, the outputs are subjected to the ReLU activation function. The final convolutional layer has 64 channel convolution filters and its outputs are subjected to the ReLU activation function. The pooling operator follows the last convolutional layer and again processes it with a 2×2 window. In pooling layers, "max-pooling" was preferred. The filter dimensions in the convolution layers are set to 3×3 and stride value is determined as "1" in all layers. The final layers are a fully connected (fc1) layer with 512 neurons and ReLu, a second fully connected layer (fc2) with 2 neurons and "softmax" and finally the classification layer. The model architecture outlined is shown in Figure 6.

All designed code platform; it is run on a workstation with 2.1 GHz, 2×10 core processor, and 32 GB memory hardware architecture. Original codes of the proposed model have been created in MATLAB/Editor Environment. The criteria used to evaluate the experimental results were chosen as the accuracy, sensitivity, specificity, false-positive rate and false-negative rate, which are widely used in the literature. The confidence score value was used as the test criterion of the RCNN model. This score shows the degree of confidence whether or not there is an object in the valid grids produced at the output of the model. The value of "0", which is the minimum, indicates that the object is absolutely absent, and that of the maximum value, "1", indicates that the object is absolutely present.

3.3. Findings

The experimental results of the CNN structure, which constitutes the backbone of the deep learning-based aircraft detection model proposed in the study, were obtained using the training and test data set detailed in Section 3.1. The test results of the designed CNN model are given in Table 2.

Table 2. Test results of the CNN model

Model	Accuracy	Sensitivity	Specificity	FP Rate	FN Rate
CNN	%98.40	%98.62	%98.13	0.0187	0.0138

The confusion matrix indicator, in which the values in Table 2 can also be easily calculated, is given in Figure 7. From the last row of the first column, percentage values of specificity and false positive rate can be easily seen. In the last row of the second column, the percentage values of the sensitivity and false-negative rates are given respectively. The value in the lower right corner of the confusion matrix is the percentage value of accuracy and classification error values. Row values in the third column of the matrix give the positive prediction and the negative prediction together with the error percentages. When the test results are examined, it can be said that the CNN network, which constitutes the main backbone of the RCNN based aircraft detection model, has achieved high-test performance.

In the images presented in Figure 8, a selection from the test results of the proposed RCNN aircraft detection model is shown. After the RCNN model was trained, tests were performed with randomly selected satellite images that were not used in the training of the system. The model outputs are the bounding box layout and the confidence score. When Figure 8 is examined, it can be seen that the bounding box placement and confidence score values of the proposed RCNN aircraft detection model are at an acceptable level. Considering all the test pictures, it can be seen that the detection model can detect almost all aircraft objects. In the satellite image containing multiple aircraft images presented in the lower-left corner in Figure 8, the bounding box placement of the

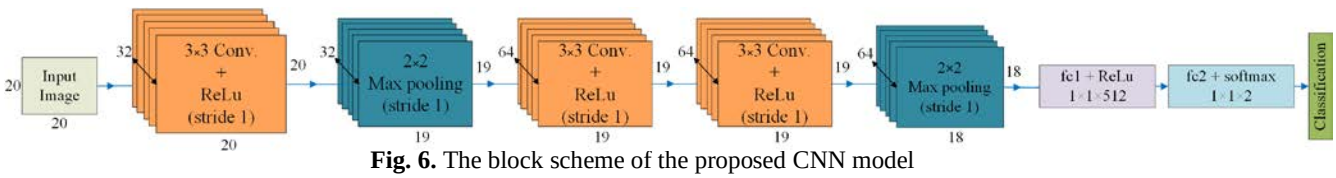

Fig. 6. The block scheme of the proposed CNN model

plane, which is very close to the grass area and corner lines, did not overlap. When all the test data are taken into consideration, it has been observed that the images with such distorting effects are detected at a high rate.

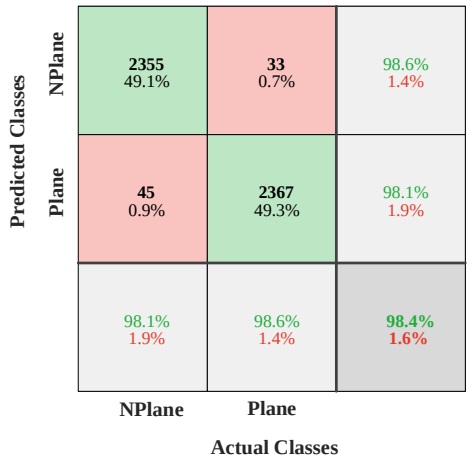

Fig. 7. Confusion matrix of the CNN testing

Fig. 8. Aircraft detection results of RCNN testing

4. Conclusion

In this study, a deep learning-based aircraft detection model has been developed. The proposed model uses the RCNN object detection operator, which uses the simple and easily applicable CNN network designed as application-specific as the backbone. The RCNN model, which performs the aircraft detection process, stands out by using the "softmax" classifier structure, unlike the basic composition, instead of using SVM classifiers. In the training and testing process of the proposed CNN model, a comprehensive dataset including "Plane" and "NPlane" classes was used. In addition, a dataset that specially created with the help of Google Earth from satellite images was used to be applied in the verification process of aircraft detection. This comprehensive training and the testing datasets were collected from airports across Turkey. Testing of the system was carried out on satellite images obtained from various airports and high-performance values were achieved. Considering that the preferred dataset of this aircraft detection model, which was prepared with a purposeful and plain structure, was not used in any other studies proposed in the literature, a table that compares the findings could not be presented because of a lack of fair comparison. In the future, studies involving aircraft detection models to be designed with advanced deep learning models are planned in a widely used publicly available aircraft dataset.

References

[1] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object Detection With Deep Learning: A Review," *IEEE Trans. Neural Networks Learn. Syst.*, vol. PP, pp. 1–21, 2019.

[2] E. DANDIL, "Bilgisayarlı Tomografi Görüntüleri Üzerinde Karaciğer Bölgesinin Bilgisayar Destekli Otomatik Bölütleme Uygulaması," *Gazi Üniversitesi Fen Bilim. Derg. Part C Tasarım ve Teknol.*, vol. 7, no. 3, pp. 712–728, 2019.

[3] W. Zhang, W. Shihao, T. Sophanyouly, J. Chen, and Y. Qian, "Deconv R-CNN for Small Object Detection on Remote Sensing Images," in *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, 2018, pp. 2491–2494.

[4] H. AÇIKGÖZ, İ. POYRAZ, and R. ÇÖTELİ, "IEEE 14-Baralı Güç Sisteminde Gerilim Kararlılığının Uç Öğrenme Makinesi İle Analizi," *Gazi Üniversitesi Fen Bilim. Derg. Part C Tasarım ve Teknol.*, vol. 7, no. 3, pp. 564–575, 2019.

[5] A. Şeker, B. Diri, and H. H. Balık, "Derin Öğrenme Yöntemleri ve Uygulamaları Hakkında Bir İnceleme," *Gazi Mühendislik Bilim. Derg.*, vol. 3, no. 3, pp. 47–64, 2017.

[6] Ö. Inik and E. Ülker, "Derin öğrenme ve görüntü analizinde kullanılan derin öğrenme modelleri," *Gaziosmanpaşa Bilim. Araştırma Derg. Gaziosmanpasa J. Sci. Res.*, vol. 6, no. 3, pp. 85–104, 2017.

[7] A. Gülcü and Z. Kuş, "Konvolüsyonel Sinir Ağlarında Hiper-Parametre Optimizasyonu Yöntemlerinin İncelenmesi," *Gazi Üniversitesi Fen Bilim. Derg. Part C Tasarım ve Teknol.*, vol. 7, no. 2, pp. 503–522, 2019.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[9] J. Li, H. C. Wong, S. L. Lo, and Y. Xin, "Multiple Object Detection by a Deformable Part-Based Model and an R-CNN," *IEEE Signal Process. Lett.*, vol. 25, no. 2, pp. 288–292, 2018.

[10] R. Girshick, "Fast R-CNN," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1440–1448.

[11] K. Lenc and A. Vedaldi, "R-CNN minus R," *Arxiv*, pp. 5.1-5.12, 2015.

[12] P. Ding, Y. Zhang, W. J. Deng, P. Jia, and A. Kuijper, "A light and faster regional convolutional neural network for object detection in optical remote sensing images," *ISPRS J. Photogramm. Remote Sens.*, 2018.

[13] V. Murugan, V. R. Vijaykumar, and A. Nidhila, "A Deep Learning RCNN Approach for Vehicle Recognition in Traffic Surveillance System," in *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*, 2019, pp. 157–160.

[14] X. Bai, C. Wang, and C. Li, "A Streampath-Based RCNN Approach to Ocean Eddy Detection," *IEEE Access*, vol. 7, pp. 106336–106345, 2019.

[15] M. Zhu, Y. Xu, S. Ma, S. Li, H. Ma, and Y. Han, "Effective airplane detection in remote sensing images based on multilayer feature fusion and improved nonmaximal suppression algorithm," *Remote Sens.*, vol. 11, no. 9, 2019.

[16] K. Zhao and X. Ren, "Small Aircraft Detection in Remote Sensing Images Based on YOLOv3," in *IOP Conference Series: Materials Science and Engineering*, 2019, vol. 533, p. 012056.

[17] C. Lee, H. J. Kim, and K. W. Oh, "Comparison of faster R-CNN models for object detection," in *International Conference on Control, Automation and Systems*, 2016, vol. 0, no. Iccas, pp. 107–110.

[18] H. Chen, J. Zhao, T. Gao, and W. Chen, "Fast airplane detection with hierarchical structure in large scene remote sensing images at high spatial resolution," in International Geoscience and Remote Sensing Symposium (IGARSS), 2018, vol. 2018-July, pp. 4846–4849.

[19] C. Yıldız and E. Polat, "Uydu Görüntülerinden Durağan Haldeki Uçakların Algılanması," in *2011 IEEE 19th Signal Processing and Communications Applications Conference, SIU 2011*, 2011, pp. 514–517.

[20] Y. Wang, A. Wang, and C. Hu, "A Novel Airplane Detection Algorithm Based on Deep CNN," in *International Conference of Pioneering Computer Scientists, Engineers and Educators ICPCSEE 2018:*, 2018, pp. 721–728.

[21] M. R. Mohammadi, "Deep Multiple Instance Learning for Airplane Detection in High Resolution Imagery," *Arxiv*, no. September, 2018.

[22] Z. Chen, T. Zhang, and C. Ouyang, "End-to-end airplane detection using transfer learning in remote sensing images," *Remote Sens.*, vol. 10, no. 1, pp. 1–15, 2018.

[23] KAGGLE, "Planes Dataset." [Online]. Available: https://www.kaggle.com/rhammell/planesnet/data.

[24] M. E. Sertkaya, "Derin Öğrenme Tekniklerinin Biyomedikal İmgeler Üzerine Uygulamaları," Fırat Universitesi, 2018.

[25] N. E. Öçer and U. Avdan, "Uzaktan Algılanmış Görüntülerde Obje Tespitinde ve Sınıflandırılmasında Derin Öğrenme Temelli Yaklaşım," in *IX. TUFUAB Teknik Sempozyumu*, 2017, pp. 140–143.

[26] M. Toğaçar, B. Ergen, and M. E. Sertkaya, "Zatürre Hastalığının Derin Öğrenme Modeli ile Tespiti," *Fırat Üniversitesi Müh. Bil. Derg.*, vol. 31, no. 1, pp. 223–230, 2019.

[27] J. Ma, J. Zhang, L. Xiao, K. Chen, and J. Wu, "Classification of Power Quality Disturbances via Deep Learning," *IETE Tech. Rev.*, vol. 34, no. 4, pp. 408–415, 2017.

[28] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective Search for Object Recognition," in *International Journal of Computer Vision*, 2013, vol. 104, no. 2, pp. 154–171.

[29] A. Nur Omeroglu, N. Kumbasar, E. Argun Oral, and I. Y. Ozbek, "Mask R-CNN Algoritması ile Hangar Tespiti Hangar Detection with Mask R-CNN Algorithm," in *2019 27th Signal Processing and Communications Applications Conference (SIU)*, 2019, pp. 1–4.

[30] MATLAB, "Get Started with the Image Labeler," 2019. [Online]. Available: https://uk.mathworks.com/help/vision/ug/get-started-with-the-image-labeler.html.