



**HAL**  
open science

# Aircraft Conflict Resolution Using Convolutional Neural Network on Trajectory Image

Md Siddiqur Rahman, Laurent Lapasset, Josiane Mothe

► **To cite this version:**

Md Siddiqur Rahman, Laurent Lapasset, Josiane Mothe. Aircraft Conflict Resolution Using Convolutional Neural Network on Trajectory Image. Ajith Abraham; Niketa Gandhi; Thomas Hanne; Tzung-Pei Hong; Tatiane Nogueira Rios; Weiping Ding. Intelligent Systems Design and Applications. 21st International Conference on Intelligent Systems Design and Applications (ISDA 2021) Held During December 13–15, 2021, 418, Springer, pp.806-815, 2022, Lecture Notes in Networks and Systems, 978-3-030-96307-1. 10.1007/978-3-030-96308-8\_75 . hal-03628435

**HAL Id: hal-03628435**

**<https://hal.science/hal-03628435>**

Submitted on 4 Apr 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Aircraft Conflict Resolution using Convolutional Neural Network on Trajectory Image

Md Siddiqur Rahman<sup>1,2,4</sup>, Laurent Lapasset<sup>2,4</sup>, and Josiane Mothe<sup>3,4</sup>

<sup>1</sup> IRIT UMR5505 CNRS, Univ.de Toulouse 1 Capitole

<sup>2</sup> DEVI, Ecole Nationale de l'Aviation Civile

<sup>3</sup> INSPE, IRIT, UMR5505 CNRS

<sup>4</sup> Univ.de Toulouse, Toulouse, France

ronicse59@gmail.com

**Abstract.** A situation between several moving aircraft is a conflict when their position is less than the internationally specified distance. To solve aircraft conflicts, air traffic controllers consider many parameters including the positioning coordinate, speed, direction, weather, etc. of the involved aircraft. This is a complex task, specifically considering the increase of the traffic. Assisting systems could help controllers in their tasks. Most conflict resolution models are based on trajectory data of a fixed number of input aircraft. Under this constraint, it is possible to resolve conflicts using machine learning models, including convolutional neuron network models. Such models cannot resolve conflicts that imply a variable number of aircraft because the input size of the model is fixed. To solve this challenge, we transformed the trajectory data into images which size does not depend on the number of planes. We developed a multi-label conflict resolution model that we named ACRnet, based on a convolutional neural network to classify the obtained images. ACRnet model achieves an accuracy of 99.16% on the training data and of 98.97% on the test data set for two aircraft. For both two and three aircraft, the accuracy is 99.05% (resp. 98.96%) on the training (resp. test) data set.

**Keywords:** Air traffic control, Convolutional neural network, Machine learning, Deep learning, Multi-label classification.

## 1 Introduction

One of the frequent tasks an air traffic control (ATC) does is to resolve conflicts between different aircraft. When two or more planes come close to each other at a certain distance, it is called a conflict. Conflict resolutions by Air Traffic Control Officer (ATCO) vary depending on the position of the aircraft at different altitudes. In this research, we tackle the en-route level (high altitude) where conflicts are resolved by changing the aircraft's heading. As the number of aircraft is increasing rapidly, providing an automated solution to conflicts would drastically help ATCOs.

Kuchar and Yang reviewed aircraft conflict detection and resolution [12] research. Although this review was conducted two decades ago, it is worth mentioning it as the authors present the basis of conflict identification and resolution,

which was mainly mathematically grounded [4, 16]. Following the other scientific domains, researchers have recently moved on to machine learning [20] including deep learning to solve aircraft conflicts [5, 6, 18].

Most models for conflict resolution are designed considering a fixed number of aircraft. Under such constraints, machine learning models where the input size cannot be changed in real-time can be applied. The resolution of conflicts where a variable number of aircraft are involved however cannot be handled and remains an important challenge. Brittain and Wei [6] applied long short-term memory (LSTM) and Zhao and Liu [21] convert trajectory data to image data to handle a variable number of aircraft in real-time using reinforcement learning. The computation of their model depends on the number of aircraft when using LSTM, and finding the perfect reward function when using reinforcement is challenging. Here, inheriting ideas from Zhao and Liu [21], we convert trajectory data to image data. The benefits of converting trajectory data to images are given below:

- (a) The trajectory of a variable number of planes can be plotted without resizing the image;
- (b) The model complexity will not change even if the number of planes changes;
- (c) Convolutional neural network (CNN) can be applied with data augmentation;
- (d) Not only can conflicts be resolved between aircraft but also between aircraft and other airspaces such as weather, military zones, etc.

To reflect the fact a conflict can be solved in different ways, we annotated each image with multi-labels, each corresponding to a possible solution. While previous research considers a single solution, we think providing several solutions is better because one solution can result into new conflicts [12].

We developed a machine learning model based on a CNN with the multi-label classification that we named ACRnet for aircraft conflict resolution CNN.

This paper is organized as follows. Section 2 discusses related work. Section 3 explains the conversion from trajectories to images. Section 4 discusses model architectures. Section 5 presents the evaluation framework. Section 6 presents the results and comparisons between different models. Finally, Section 7 concludes this paper.

## 2 Related work

Most of the initial research in conflict detection and resolution was based on mathematics [2, 4, 7, 16]. According to Pham *et al.* [14], it is essential for these models that the information is noise-free and accurate positioning, which is not possible because trajectory information comes from the approximate combined results of different surveillance radars. Recent research has thus focused rather on machine learning [1, 5, 9, 14, 20].

Many methods were proposed in which all the flights are assumed to be allowed for free routing or free-flight and that the planes share their information

from the onboard system [1, 2, 9]. In this case, the pilots of all the aircraft involved in the conflict can resolve the conflict by communicating within themselves without the help of ATCO. Although it is interesting, in reality, its application is still a long way off.

Since annotated ATC data is difficult to obtain, many methods have been proposed based on reinforcement learning to resolve aircraft collisions. Brittain and Wei in [5] applied two-level deep reinforcement learning in a hierarchical way while Pham *et al.* [14, 15] proposed a single agent-based deep reinforcement learning model for two aircraft at the same altitude. Zhao and Liu [21] also applied reinforcement learning and CNN using image data where each image contains the current position of aircraft associated with the conflict. However, a proper reward function is required for the best resolution of reinforcement learning and finding that function is the most challenging task because it depends on features like certain rules and conditions.

There are very few supervised learning applications for aircraft collision detection and resolution [10, 17, 20]. Recently, Rahman *et al.* [13, 17] introduced a fully supervised deep neural network (NN) model based on simulated data. The advantages of using simulated data are that many variations are possible with simulation data that may not be obtained from actual data. Kim *et al.* [10] considered two different supervised models, one based on a NN and the other based on multiple support vector machines (SVM). Both are based on multi-class classification where the best output is selected based on the highest probability score. The NN model for horizontal (resp. vertical) conflict achieved 71.3% (resp. 93.1%) of accuracy, while the SVM-based model for horizontal (resp. vertical) conflict achieved 72.2% (resp. 93.3%) of accuracy. CRMLnet model in [18] is similar but for multi-label classification (multiple resolutions for one conflict) using the last 5-minute of trajectories (series of locations) of all the aircraft associated with the conflict. This model only solves the horizontal conflict at the en-route level and achieved an accuracy of 98.72% and ROC of 0.999. The main remaining challenge is that there is no supervised learning method that can handle a variable number of aircraft.

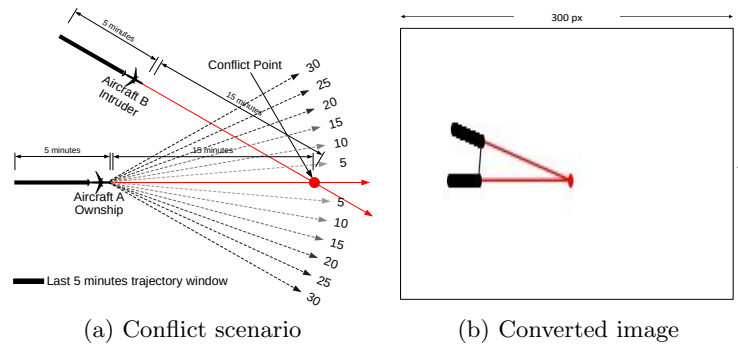
By combining both ideas of 5-minute trajectory and image data [18, 21], we suggest here a multi-label classification model based on a CNN where each image contains a 5-minute trajectory for each aircraft associated with the conflict. The main purpose of this research is not to find the best image processing model but to easily overcome many challenges of existing sequence-based models through image processing with higher performance to resolve the aircraft conflict.

### 3 Data

We used two different datasets in the evaluation part (a) initial trajectory data from [18] and (b) converted trajectories into images.

**(a) Initial trajectory data:** Each conflict sample of initial data contains the last 5-minute of the trajectory (a series of positions) for each aircraft associated with the conflict. The resolution can be made for each conflict by changing the

ownship's (see Figure 1 (a) aircraft A heading from 30 degrees left to 30 degrees right by 5 degrees (total 12 heading directions). Each conflict is associated with multiple heading resolutions (multi-labels) for a single conflict. The position of each aircraft changes every 10 seconds following a 5-minute trajectory. Trajectory data contains 9 features: time, latitude (aircraft A and B), longitude (aircraft A and B), altitude (aircraft A and B), heading (aircraft A and B). These 9 features repeat every 10 seconds in the 5 minutes (300 seconds). The total parameters for 5 minutes including an angle between the aircraft is:  $271 (1 (\text{angle}) + 9 (\text{features}) \times 30 (300 \text{ sec} \div 10 \text{ sec}))$ .



**Fig. 1. Aircraft conflict trajectory with heading resolutions and conversion into the image.** (a) the black solid line just behind the plane represents the last 5 minutes of trajectory before the conflict is detected. All dotted lines show possible heading changes to resolve the conflict (b) plots the positioning coordinate of the last 5 minutes where the black line is the current distance between aircraft and red lines are the distance between aircraft and the conflict point.

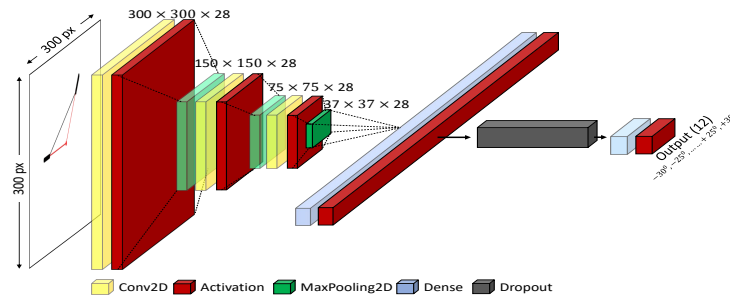
**(b) Conversion of trajectories into images:** Using trajectory data to resolve conflicts, the number of input parameters depends on the number of aircraft. Converting trajectories into images helps to keep the model input size unchanged even if the number of planes changes. The model can then be applied whatever the number of planes is.

In Figure 1, (a) shows a conflict scenario with the last 5 minutes of trajectories and (b) shows the image that converted from trajectory indicating the distance from the conflict point to the aircraft (red lines). The thin black line indicates the current distance between aircraft; it is the last point of the aircraft before the conflict is detected. The image size is  $300px \times 300px$ , fixed. It is possible to plot trajectories of multiple planes without changing the size of the image. Thus, there will be no change in the input size if the number of aircraft changes at run time. We applied image augmentation to increase the volume of the training data. Typically, image augmentation includes rotation, shifting, zooming, flipping, etc. We applied rotation only. We cannot apply some of the usual data augmentations because the resulting samples may be labeled differently. We created a total of

1,656 image samples, of which 1,516 images contain two aircraft and 140 contain three aircraft.

#### 4 ACRnet: Aircraft conflict resolution CNN model

We focused on two different supervised models: (a) CRMLnet, a model based on a NN [18] and (b) ACRnet, the model based on CNN we proposed. CRMLnet model consists of three layers: input, output, and a hidden layer. Input and hidden layers are of the same size, 271 nodes (neurons), as the number of trajectory parameters while output layer is 12. For ACRnet, the performance of a CNN model depends on its hyper-parameters such as the number of convolutional layers, the number of nodes in each layer, percentage of dropout, different activation functions, etc. We use random search [3] algorithm to choose the hyperparameter values.



**Fig. 2. ACRnet: Aircraft conflict resolution CNN model.** The size of the first convolutional layer (Conv2D) is  $300 \times 300$  with 28 nodes (filters) as the image size is  $300px \times 300px$ . This model contains 3 hidden layers and each hidden layer (Conv2D) of this model has 28 nodes (filters). The activation function is ReLU except for the output layer that is sigmoid. Finally, there are 12 nodes in the output layer.

We designed ACRnet as a simpler model than the other image processing model, with a lower number of layers and nodes, to avoid overfitting given the relatively small number of samples we have. Figure 2 shows the architecture of the model. We chose the widely used rectified linear units (ReLU) activation function (Figure 2, red blocks), for all the hidden layers. Several MaxPooling layers with size  $2 \times 2$  are used to reduce the dimension of each Conv2D layer by half. All filter sizes are halved from the previous layer to the next layer. After the final Conv2D, there is a Dense layer or fully connected layer that converts the final Conv2D into a 1D vector. Right after this dense activation layer (ReLU), there is a Dropout layer to exclude the less important information (in this case, we exclude 50%). Finally, there are 12 nodes in the output layer that provide binary decisions for 12 headings described in Figure 1 (a) using the sigmoid activation function.

## 5 Evaluation framework

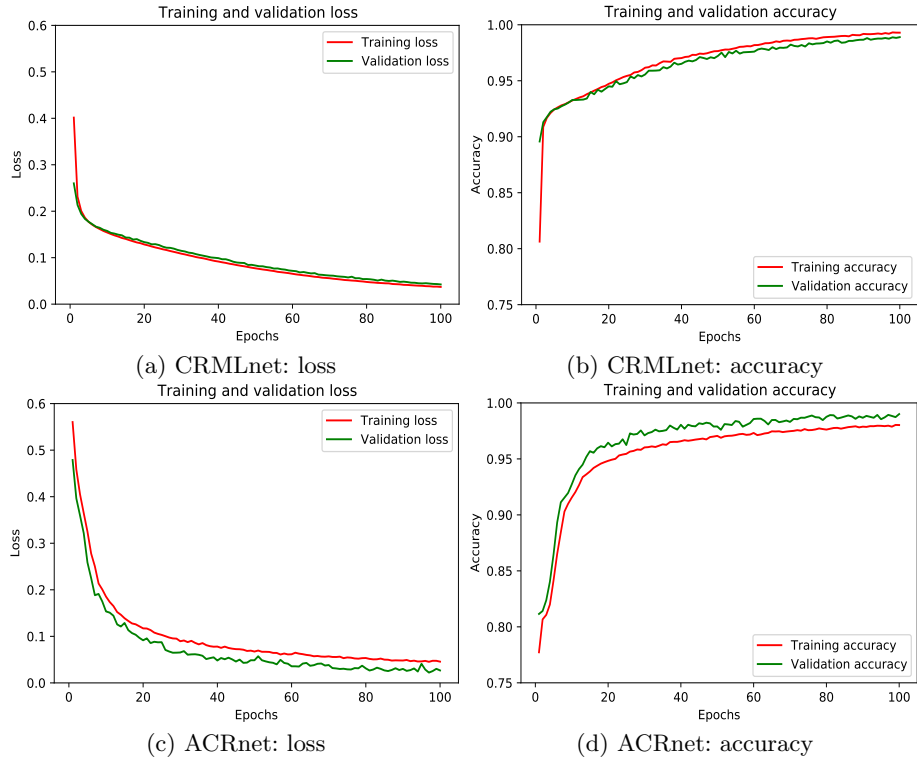
We use 10-fold cross-validation [11]. We trained the CRMLnet model with trajectory data and the ACRnet model with image data (see Section 3) using up to 100 epochs, where 1 epoch means a complete cycle of training data passes through the model network. We keep 15% of test data separate from the 10-fold cross-validation training. We also applied two widely used image classification models: VGG16 [19] and ResNet [8] to compare with the ACRnet model. For both VGG16 and ResNet, we only resize the input layer to  $300px \times 300px$  and the output layer to 12 nodes of the initial model. Activation functions are the same for all the models in this paper as in ACRnet (Section 4). We used the usual performance measures as follows:

Accuracy (Acc) is the total number of correctly classified samples over the total number of actual inputted sample. Sensitivity( $S_n$ ) also known as recall corresponds to the true positive rate. Specificity ( $S_p$ ) presents the true-negative rate. Precision also called positive predictive value (PPV) is the percentage of correctly classified positive samples among all positive predictions.  $F_1$  score combines precision (PPV) and recall (sensitivity).

False-positive rate (FPR) is the percentage of false-positive (FP) prediction among the total number of negative ground truth (FP + TN). False-negative rate (FNR) is the proportion of false-negative (FN) prediction among the total number of positive ground truth (FN + TP). Matthews correlation coefficient (MCC) is usually used for binary classification. It ranges between -1 and 1. MCC is 1 if FP = FN = 0, TP  $\neq$  0 and TN  $\neq$  0 which means there is no incorrectly classified sample. MCC is -1 if TP = TN = 0, FP  $\neq$  0, and FN  $\neq$  0 which means the model cannot correctly classify a single sample.

## 6 Results and Discussion

Figure 3 (a) and Figure 3 (b) show the loss and the accuracy curve respectively of the CRMLnet model for 10-fold cross-validation. Figure 3 (c) and Figure 3 (d) show the loss and the accuracy curve for the ACRnet model. In both cases, the x-axis represents the number of epochs used to train the model up to 100. The y-axis in Figure 3 (a) and (c) show the losses in between the range of 0.0 and 0.6 and Figure 3 (b) and (d) show the accuracy. In parallel with the visual loss and accuracy, Table 1 reports the test and validation results of the models using 10-fold cross-validation. In this table, Block 1 shows the ACRnet model’s score for two different datasets. ACRnet<sub>2</sub> is the scores using image data containing two aircraft while ACRnet<sub>3</sub> is the scores using images containing both two and three aircraft. For both two and three aircraft, the ACRnet model architecture remained unchanged. Block 2 represents the performance score of the CRMLnet model using trajectory data while Block 3 shows the performance score of VGG16 [19] and ResNet [8] using the same image data as ACRnet. According to loss and accuracy functions in Figure 3, CRMLnet looks a little better than ACRnet but the scores, especially the highlighted scores, in Table 1 shows ACRnet is much



**Fig. 3. Up to 100 epochs both CRMLnet and ACRnet models performed well without any overfitting.** The comparison between losses (a) & (c) and accuracy (b) & (d) shows that the performances of both models are very close to each other while CRMLnet is a little better than ACRnet comparing the training and validation curve.

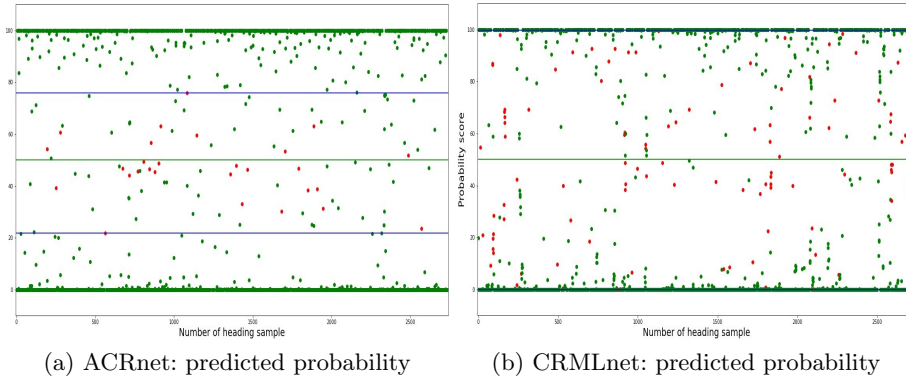
better than others. So, using image data not only we handle a variable number of aircraft but also we increase the performance of the model. Table 1 Block 3 shows the scores of VGG16 and ResNet; they do not perform well compare to ACRnet. This may be due to their many layers. If there were more samples, the performance of those two models may be better. The main purpose of our research was not to find the best image processing model but to easily overcome the challenge of existing sequence-based models through image processing with higher performance.

We further tested ACRnet and CRMLnet to see the probability score of each heading decision and to know how image data helps in increasing performance. Figure 4 (a) and (b) plot the predicted probability scores of all the individual heading resolutions for ACRnet and CRMLnet. The green horizontal line in the middle of both figures (a) and (b) indicate 50% of the probability score line exactly. All positive classes ('1') are above the green horizontal line and all negative classes ('0') are below that line. The green dots represent the samples



**Table 1. ACRnet performs much better than CRMLnet, VGG16, and ResNet.** Here, the 1<sup>st</sup> column corresponds to the models' name. The 2<sup>nd</sup> column is the test accuracy of models. All subsequent columns are: accuracy (Acc), area under receiver operating characteristic curve (auROC), area under precision-recall (auPR), specificity( $S_p$ ), sensitivity ( $S_n$ ), positive predictive value (PPV), false-negative rate (FNR), false-positive rate (FPR), Mathew's correlation coefficient (MCC), and  $F_1$  score. Block 1 shows the ACRnet model's score for two aircraft (ACRnet<sub>2</sub>) and mixed (two and three) aircraft (ACRnet<sub>3</sub>). The highlighted scores (Block 1) are the most significant where ACRnet is much better than CRMLnet (Block 2), VGG16 (Block 3), ResNet(Block 3).

Model	Test					Validation					
	Acc	Acc	auROC	auPR	$S_p$	$S_n$	PPV	FNR	FPR	MCC	$F_1$
ACRnet <sub>2</sub>	<b>98.97%</b>	<b>99.16%</b>	<b>0.999</b>	0.999	99.41%	98.66%	98.82%	1.34%	0.59%	<b>0.981</b>	<b>0.987</b>
ACRnet <sub>3</sub>	<b>98.96%</b>	<b>99.05%</b>	<b>1.000</b>	0.999	99.20%	98.78%	98.63%	1.22%	0.80%	<b>0.980</b>	<b>0.987</b>
<b>Block 2</b>											
CRMLnet	96.38%	98.76%	0.999	0.999	99.20%	97.87%	98.40%	2.13%	0.80%	0.972	0.981
<b>Block 3</b>											
VGG16	79.97%	80.93%	0.771	0.607	88.93%	65.05%	74.78%	34.95%	11.07%	0.561	0.694
ResNet	92.79%	91.34%	0.973	0.951	92.30%	89.44%	85.42%	10.56%	7.70%	0.809	0.874



**Fig. 4. ACRnet is more confident in predicting using image data than CRMLnet with trajectory data.** Here x-axis of both (a) and (b) present the total number of heading decisions ( $288$  (test conflict sample)  $\times$   $12$  (heading degree) =  $2736$ ) and the y-axis is the probability between 0% and 100%. All the dots above the green line in the middle (threshold = 50%) are the positive class ('1') and below are the negative class ('0'). All red dots are incorrectly classified that are bounded by the blue lines while green dots are correctly classified. The distance between blue lines (incorrectly classified boundaries) is shorter for ACRnet than it is for CRMLnet. The blue lines are overlapping on the 0% and 100% scoreline for CRMLnet. It means there are some incorrectly classified samples that are closed to 0% and 100%. Thus, ACRnet is more confident than CRMLnet because the shorter the distance between the blue lines, the more confident the model is.

that are correctly classified while the red is for misclassified. All misclassified samples (red-colored) are fallen inside the blue line boundaries. For both (a) and (b) in Figure 4, the x-axis presents the total number of heading degrees ( $288$  (test conflict sample)  $\times 12$  (heading degree) =  $2736$ ) and the y-axis presents the probability score of each heading resolution between 0% and 100%. The blue boundary (misclassified) for CRMLnet is overlapped on the 0% scoreline and 100% scoreline, so, it is invisible in Figure 4 (b) while it is clearly visible for ACRnet in Figure 4 (a). This means that CRMLnet classifies many samples incorrectly but with high confidence. Thus, ACRnet is more confident than CRMLnet because the shorter the distance between the blue lines, the more confident the model is.

## 7 Conclusion

This research aims to provide a generalized model to resolve a conflict where a variable number of aircraft are involved. Two things were considered: (a) create image data from trajectory sequence (b) find a CNN-based model to classify these images. We defined a relatively small CNN-based model, ACRnet, and found an accuracy of 98.97% and 98.96% for the conflict resolution classification of two and mixed (both two and three) aircraft. We compared our ACRnet model with the CRMLnet [18] model using 10-fold cross-validation where the input data of ACRnet are images and of CRMLnet are trajectory sequences. We also compared ACRnet with two other widely used CNN-based models: VGG16 [19] and ResNet [8]. Overall, ACRnet performs much better than the other models. The main purpose of this research was not to find the best image processing model, but rather to show that the use of image data not only overcomes the input dimension problem but also increases the performance of the model. An interesting aspect of using image data is that it can be used not only for conflict between planes but can also contain other information without changing the image size such as weather, thunderstorms, military zones, etc. Thus, our future research will include that information for better resolution of aircraft conflicts as well as more than 3 aircraft.

## References

1. Alam, S., Shafi, K., Abbass, H.A., Barlow, M.: An ensemble approach for conflict detection in free flight by data mining. *Transportation research part C: emerging technologies* 17(3), 298–317 (2009)
2. Alonso-Ayuso, A., Escudero, L.F., Olaso, P., Pizarro, C.: Conflict avoidance: 0-1 linear models for conflict detection & resolution. *Top* 21(3), 485–504 (2013)
3. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. *The Journal of Machine Learning Research* 13(1), 281–305 (2012)
4. Bilimoria, K.: A geometric optimization approach to aircraft conflict resolution. In: 18th Applied aerodynamics conference. p. 4265 (2000)

5. Brittain, M., Wei, P.: Autonomous aircraft sequencing and separation with hierarchical deep reinforcement learning. In: Proceedings of the International Conference for Research in Air Transportation (2018)
6. Brittain, M.W., Wei, P.: One to any: Distributed conflict resolution with deep multi-agent reinforcement learning and long short-term memory. In: AIAA Scitech 2021 Forum. p. 1952 (2021)
7. Erzberger, H., Paielli, R.A., Isaacson, D.R., Eshow, M.M.: Conflict detection and resolution in the presence of prediction error. In: 1st USA/Europe Air Traffic Management R&D Seminar, Saclay, France. pp. 17–20. Citeseer (1997)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
9. Jiang, X.r., Wen, X.x., Wu, M.g., Wang, Z.k., Qiu, X.: A svm approach of aircraft conflict detection in free flight. *Journal of Advanced Transportation* 2018 (2018)
10. Kim, K., Hwang, I., Yang, B.J.: Classification of conflict resolution methods using data-mining techniques. In: 16th AIAA Aviation Technology, Integration, and Operations Conference. p. 4075 (2016)
11. Kohavi, R., et al.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *Ijcai*. vol. 14, pp. 1137–1145. Montreal, Canada (1995)
12. Kuchar, J.K., Yang, L.C.: A review of conflict detection and resolution modeling methods. *IEEE Transactions on intelligent transportation systems* 1(4), 179–189 (2000)
13. Lapasset, L., Rahman, M.S., Mothe, J.: Solving aircraft conflicts: data resources. In: 1st International Conference on Cognitive Aircraft Systems (ICCAS 2020). p. 76 (2020)
14. Pham, D.T., Tran, N.P., Alam, S., Duong, V., Delahaye, D.: A machine learning approach for conflict resolution in dense traffic scenarios with uncertainties (2019)
15. Pham, D.T., Trant, N.P., Goh, S.K., Alam, S., Duong, V.: Reinforcement learning for two-aircraft conflict resolution in the presence of uncertainty. In: 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF). pp. 1–6. IEEE (2019)
16. Prandini, M., Lygeros, J., Nilim, A., Sastry, S.: A probabilistic framework for aircraft conflict detection. In: Guidance, Navigation, and Control Conference and Exhibit. p. 4144 (1999)
17. Rahman, M.S.: Supervised machine learning model to help controllers solving aircraft conflicts. In: ADBIS, TPD and EDA 2020 Common Workshops and Doctoral Consortium. pp. 355–361. Springer (2020)
18. Rahman, M.S., Lapasset, L., Mothe, J.: Multi-label classification of aircraft heading changes using neural network to resolve conflicts. *arXiv preprint arXiv:2109.04767* (2021)
19. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014)
20. Srinivasamurthy, A., Motlicek, P., Singh, M., Oualil, Y., Kleinert, M., Ehr, H., Helmke, H.: Iterative learning of speech recognition models for air traffic control. In: *Interspeech*. pp. 3519–3523 (2018)
21. Zhao, P., Liu, Y.: Physics informed deep reinforcement learning for aircraft conflict resolution. *IEEE Transactions on Intelligent Transportation Systems* (2021)