Aircraft Recognition in Remote Sensing Images Based on Artificial Neural Networks

Muhammad Fauzan Abrar[1], Vina Ayumi[2]

[1]Universitas Mercu Buana, Jl Meruya Selatan, No 1, Jakarta Barat 11650, Indonesia
[1]41518010158@student.mercubuana.ac.id; [2] vina.ayumi@mercubuana.ac.id
*corresponding author

ARTICLE INFO

Article History:
Received January 5, 2022
Revised October 14, 2023
Accepted October 25, 2023

Keywords:
Computer Vision
Artificial Neural Network
Convolutional Neural Network
Residual Neural Network

Correspondence:
E-mail:
41518010158@student.mercubuana.ac.id

ABSTRACT

Computer vision (CV) is a field of artificial intelligence (AI) that enables computers and systems to obtain data from images, recordings, and other visual information sources. Image recognition, a subcategory of computer vision, addresses a bunch of strategies for perceiving and taking apart pictures to engage the automation of a specific task. It is suitable for perceiving places, people, objects, and various types of parts inside an image and reaching deductions from them by analyzing them. With these kinds of utilities, it is a no-brainer that computer vision has its use cases in the military world. Computer vision can be immensely useful for intelligence, surveillance, and reconnaissance (ISR) work. This paper provides information on how computer vision might be used in ISR work. This paper utilizes artificial neural networks (ANN) such as convolutional neural networks (CNN) and residual neural networks (ResNet) for demonstration purposes. In the end, the ResNet model managed to edge out the CNN model with a final validation accuracy of 90.9% compared to a validation accuracy of 86% on the CNN model. With this, computer vision can help enhance the efficiency of human operators in image and video data-related work.

1. Introduction

Computer vision (CV) is a field of artificial intelligence (AI) that enables computers and systems to obtain data from images, recordings, and other visual information sources. If AI is interpreted as equipping computers to think, CV, on the other hand, can be interpreted as enabling computers to observe, see, and recognize. It is safe to say that the goal of CV is to create an autonomous system that can perform the endeavors of the human visual system or even surpass them.

Image recognition, a subcategory of computer vision, addresses a bunch of strategies for perceiving and taking apart pictures to automate a specific task. It is suitable for perceiving places, people, objects, and various types of parts inside an image and reaching deductions from them by analyzing them. An example of image recognition is Google Lens. By using a device’s camera to capture images and send out relevant information about the object that it manages to identify.

With these kinds of utilities, it is a no-brainer that computer vision has its use cases in the military world. Computer vision can be immensely useful for intelligence, surveillance, and reconnaissance (ISR) work. Computer vision can help enhance the efficiency of operators in image and video data-related work. Thereby increasing their capacity to pursue other higher-value lines of work. Many militaries around the world are heavily investing in computer vision for intelligence work, such as detecting opposing countries' military hardware. These are the need for autonomous operation and the need to make greater use of the outputs from a diverse range of sophisticated sensors [1].

These are some instances of utilizing computer vision in a military setting. Computer vision can be used to recognize military vehicles based on the images that are posted on social media [2]. And other instances of computer vision used to recognize infantry fighting vehicles (IFV) and tanks [3] [4] [5]. Ground vehicles are not the main interest of this paper. This paper focuses on aircraft recognition in remote sensing images utilizing CNN and ResNet. Harnessing the power of artificial

http://dx.doi.org/10.36596/jcse.v4i2.381

 e-ISSN 2721-0251

neural networks (ANN) for image recognition stems from the intricate demands of identifying aircraft in satellite imagery. This study leverages the capabilities of convolutional neural networks (CNN) and residual neural networks (ResNet) to process data and derive precise classifications for diverse aircraft types captured in satellite images.Based on previous research by An Zhao et al. [6], aircraft type recognition is critical both in civilian and military use cases. It is said that if research is able to be implemented well enough, it will help alleviate the work of human operators. Qichang Wu et al. [7] also share the same sentiment. They added that aircraft recognition is not only necessary but also challenging. They stated that aircraft recognition is different from other natural object recognition because: (i) the number of aircraft types is limited; and (ii) each type of aircraft has a fixed size and shape. Maybe TNI AU (Indonesian Air Force) could invest in such technology if they haven't already. It could help with assessing hostile countries' battle forces so that they can come up with appropriate countermeasures to combat those forces. Since generally, computer vision can be developed quickly and at a low cost, it is a perfect fit for the military. Though, unlike for general use, obtaining datasets for military applications can be quite challenging since other countries don't want to share images of their military hardware; doing so could endanger national security.

2. Method

The Keras utility library was used in this study. This relates to scenarios involving the recognition of images using artificial neural networks (ANN). The study processes data using convolutional neural networks (CNN) and residual neural networks (ResNet) to classify different aircraft types using satellite photos. In order to generate a prediction of particular conditions based on the gathered dataset, the paper gives priority to image recognition—a novel approach that may prove useful for various use cases. The data collection phase involves acquiring the dataset through the internet. Ensuring the correct labels are affixed to the correct images is known as pre-processing data. Assign the appropriate label to the data if there are any anomalies. Pre-processing is necessary because numerous miss-labelings have been discovered throughout the dataset. The next step is to manually give them the right label for their retrospective after sorting. The data must then be divided into two folders. The folders for the train and test datasets. Data augmentations on the meticulously reorganized dataset are the next step. The specified ANN models, CNN and ResNet, are then used to implement the rearranged data. The accuracy levels are then evaluated through testing. Are the correct tags properly implemented or not. Then determine from the results of the two architectures which is more suited for this specific test. The design phases are provided by Figure 1, which visualizes the ResNet design process, and Figure 2, which visualizes the CNN design process.

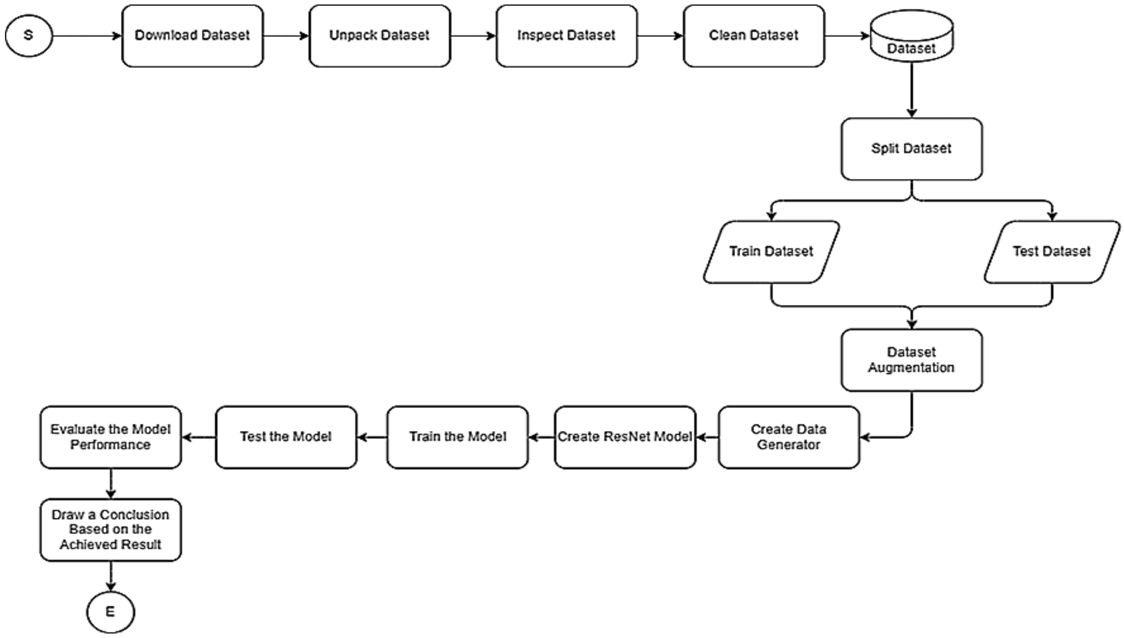

Figure 1. CNN Design Process

http://dx.doi.org/10.36596/jcse.v4i2.381

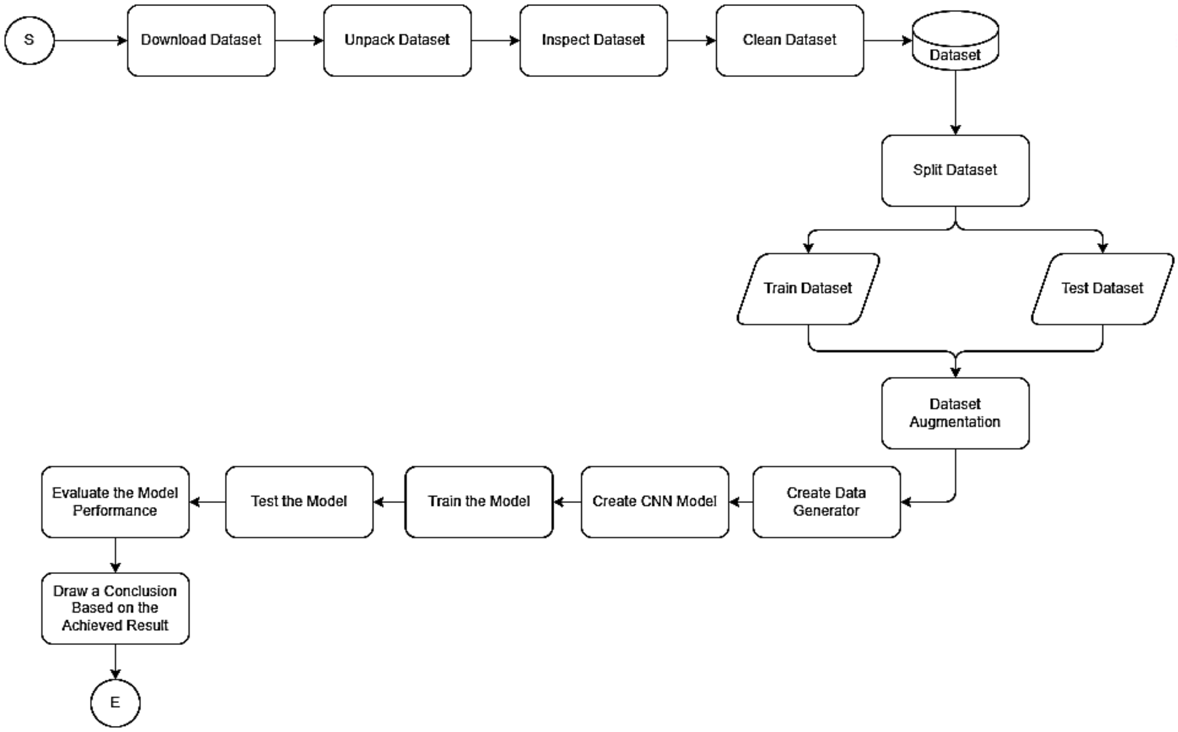

Figure 2. ResNet Design Process

2.1 Dataset

As an option to obtain training data from freely accessible sources, the "Muti-type Aircraft of Remote Sensing Images: MTARSI [8] dataset is used. The dataset contains 9,385 images of 20 different aircraft models. Though at first it seems that the dataset is organized, in fact it is not. There are multiple cases of false class labeling. the sample dataset shown in figures 3–13.

Twin-prop engined light aircraft labelled as U-2

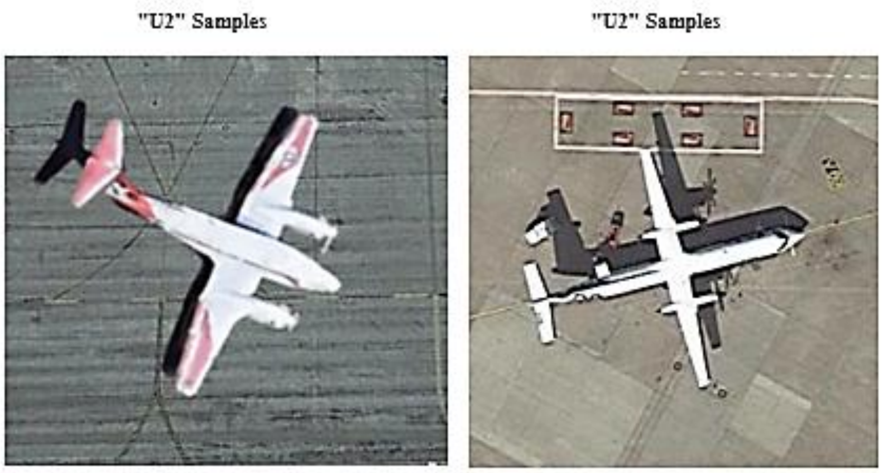

Figure 3. “U-2” Samples

http://dx.doi.org/10.36596/jcse.v4i2.381

C-5 and C-17 got mixed up and vice versa, also Airliner labelled as C-5

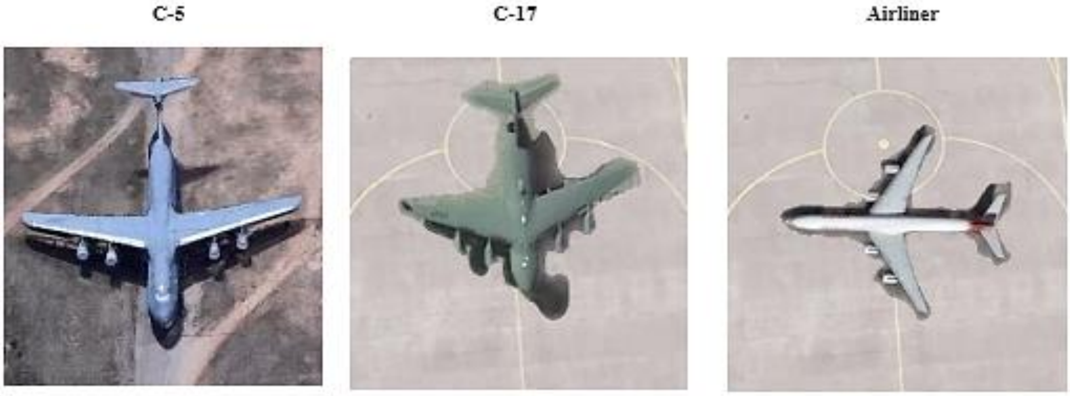

Figure 4. Samples for C-5, C-17, and Airliner

Light Aircrafts labelled as A26 and P63

Figure 5. “A26” and “P63” with Actual A26 and P63
Source: Ragnhild and N. Crawford, Douglas A26 Invader. 2016. G. Goebel, Bell P-63 Kingcobra, Chino, California. 2007.

http://dx.doi.org/10.36596/jcse.v4i2.381

 e-ISSN 2721-0251

B29 labelled as P3

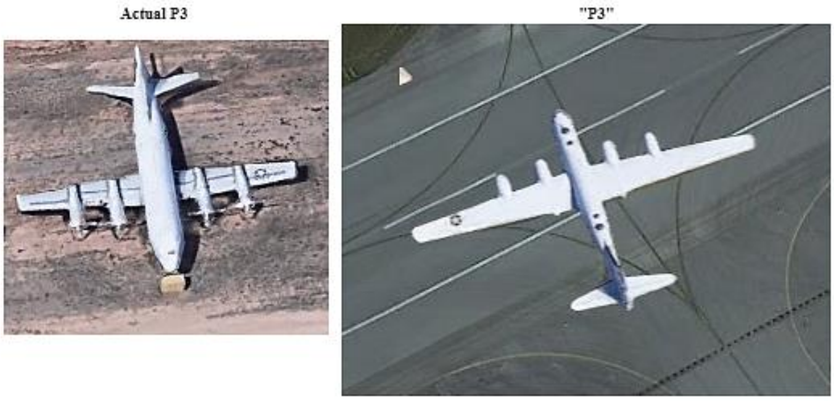

Figure 6. P3 Samples

P3, B29, and E3 labelled as C-130

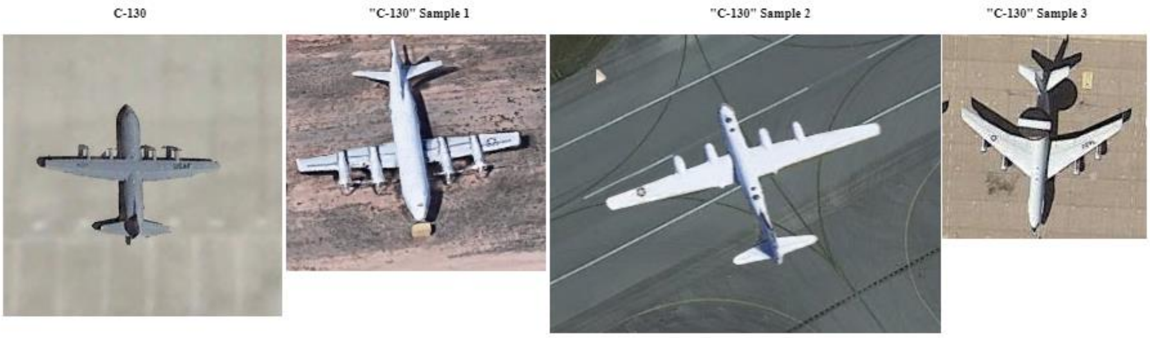

Figure 7. C-130 and it's "Samples"

E2 labelled as E3

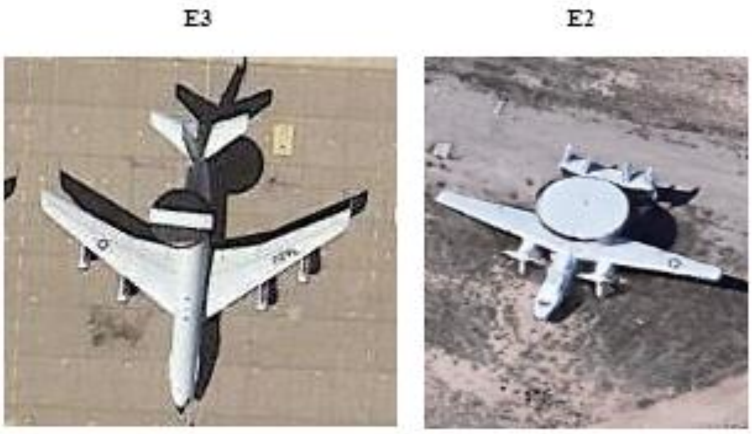

Figure 8. E3 Samples

http://dx.doi.org/10.36596/jcse.v4i2.381

All aircraft that have fuselage mounted engines are labelled as "C-21"

Figure 9. C21 Samples

C-135 and Boeing 707 labelled as KC-10

Figure 10. KC-10 Samples

Sukhoi Flanker series(?) and F-15 labelled as F-16

Figure 11. F-16 Samples

http://dx.doi.org/10.36596/jcse.v4i2.381

F-4, F-15, F-16, F-18, and F-35 being labelled as F-22

Figure 12. F-22 Samples

Aircrafts that does not need to be edited

Figure 13. Samples of Aircrafts That Does Not Have False Labelling

Figure 3 displays samples from the MTARSI dataset where twin-prop engined light aircraft are inaccurately labeled as "U-2." The mislabeling trend continues in Figure 4, where C-5 and C-17 aircraft are mixed up, and an Airliner is erroneously labeled as C-5. Moving forward to Figure 5, light aircrafts are mistakenly labeled as A26 and P63, with a reference to the actual A26 and P63 provided from sources by Ragnhild and N. Crawford, and G. Goebel, respectively. Figure 6 showcases instances where B29 aircraft are incorrectly labeled as P3, while Figure 7 reveals confusion among P3, B29, and E3, all labeled as C-130. Figure 8 demonstrates the mislabeling of E2 aircraft as E3. Intriguingly, Figure 9 highlights a curious pattern where all aircraft with fuselage-mounted engines are uniformly labeled as "C-21." Figure 10 captures the mix-up between C-135 and Boeing 707, both mistakenly labeled as KC-10. Figure 11 sheds light on the misclassification of

http://dx.doi.org/10.36596/jcse.v4i2.381

Sukhoi Flanker series(?) and F-15 as F-16. Figure 12 further compounds the confusion, featuring F-4, F-15, F-16, F-18, and F-35 aircraft all labeled as F-22. Concluding the series, Figure 13 provides relief with samples of aircraft that do not suffer from false labeling, offering a glimpse of the dataset's integrity amidst the intricacies uncovered in the preceding images.

Following a meticulous process of manual reorganization, the dataset now encompasses a comprehensive collection of 22 distinct classes, resulting in a total of 9,146 images after pruning. This curated dataset captures a diverse array of aircraft, each belonging to one of the following 22 classes:

Figure 14. Dataset After Assigning Correct Labels

The dataset consists of a total of 9,146 images, and it has been divided into two subsets for distinct purposes. A majority of the images, precisely 7,330, have been allocated for the training phase. This substantial subset is crucial for the model to learn and generalize patterns from the data. On the other

http://dx.doi.org/10.36596/jcse.v4i2.381

hand, the remaining 1,816 images have been set aside for testing. This separate testing subset is invaluable for evaluating the model's performance on unseen data, providing a reliable measure of its effectiveness and generalization capabilities. This division of the dataset into training and testing sets ensures a robust assessment of the model's proficiency in recognizing patterns and making predictions.

2.2 Data Augmentation

Data augmentation has been shown to produce promising ways to increase the accuracy of classification tasks[9][10]. Data augmentation is a key technique of machine learning. It consists in increasing the number of data, by artificially synthesizing new samples from existing ones. In order to make the model more robust, prevent overfitting and enable it to generalize better data augmentation techniques were utilised[11]. The ImageDataGenerator in Keras is used to augment the datasets at hand, with the parameters of the augmentation provided in the table 1.

Table 1. Data Augmentation with parameters

No	Data Augmentation	
	Augments	***Parameters***
1	Re-Scale	1./255
2	Shear	0.2
3	Zoom	0.2
4	Horizontal Flip	True

How the images looked like before and after augmentation, shown ini figure 15.

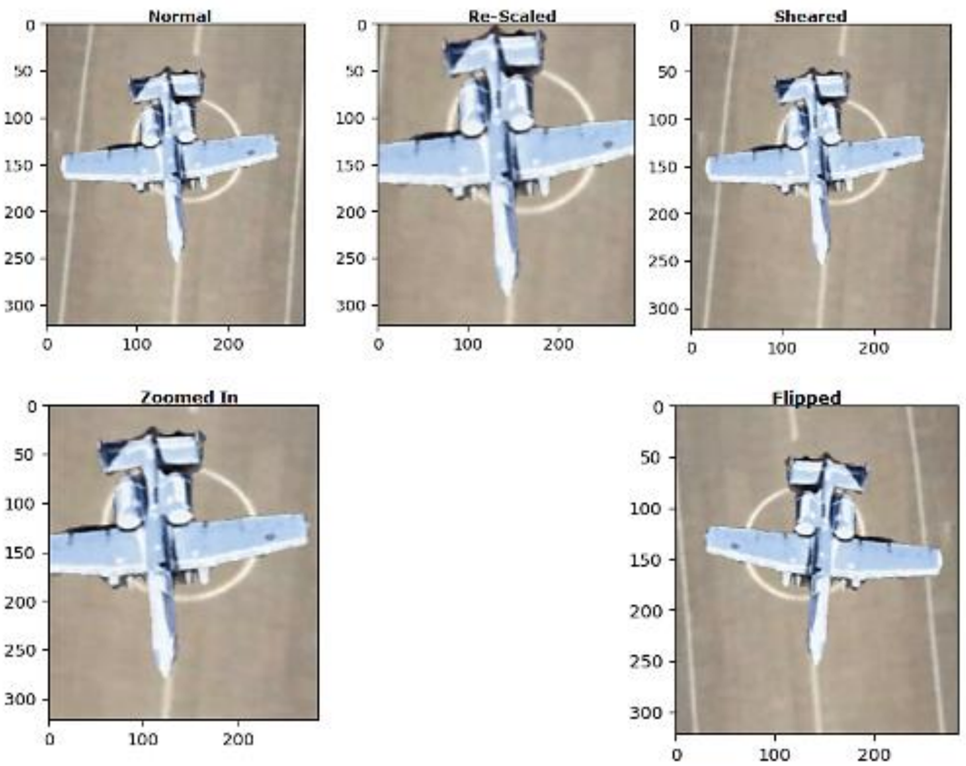

Figure 15. Dataset Augmentation Parameters

There is all to it for the preprocessing process of the datasets before we can feed it into the models.

2.3 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a class of deep neural networks, most usually applied to investigate visual imagery[12] It found great success with researchers. As stated by Liu S and Liu

http://dx.doi.org/10.36596/jcse.v4i2.381

Z[13] Normally, a CNN involves a pile of convolutional and pooling layers. The convolutional layer can create feature maps by convolving the input feature maps or image with a set of learnable kernels. What's more the pooling layer can pool data of a given district on output feature maps in order to achieve down sampling and expansion of the receptive field.

2.4 Residual Neural Network (ResNet)

Residual Neural Network (RNN or ResNet) is a type of neural network that was introduced in 2015 by Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun in "Deep Residual Learning for Image Recognition"[14]. ResNet was presented later after CNN. Extra layers are added to a DNN to further improve accuracy and performance and are helpful in solving intricate problems. This problem of training very deep networks has been eased with the presentation of ResNet. Typical ResNet models are carried out with double- or triple- layer skips that contain nonlinearities (ReLU) and batch normalization in between[15].

3. Results and Discussion

Table 1, presents the key parameters employed for both the Convolutional Neural Network (CNN) and ResNet models. These parameters are critical configurations that influence how the models are trained and evaluated.

Table 2. Model Parameters

No	Model Parameters	
	Parameters	***Value***
1	Batch Size	32
2	Optimizer	Adam
3	Image Size	70
4	Metric	Accuracy
5	Class Mode	Categorical
6	Epoch	50

Moving on to the evaluation metrics of the end results for the CNN model and ResNet model. First, the CNN model. Provided below are the Training Accuracy v Validation Accuracy Chart, Training Loss v Validation Loss Chart, End Training Accuracy, End Training Loss, End Validation Accuracy, and End Validation Loss.

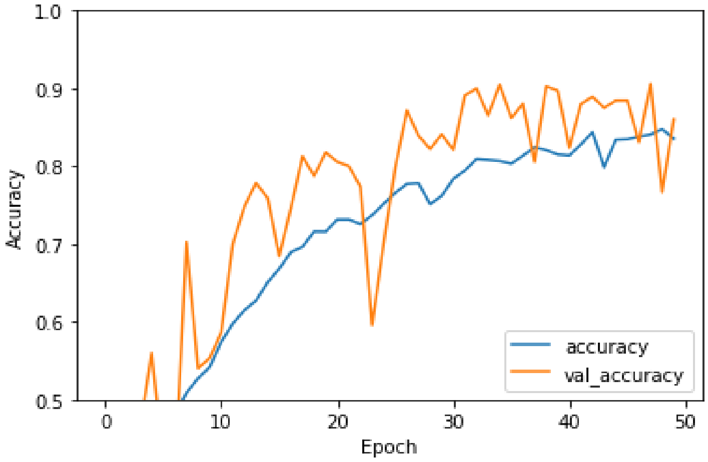

Figure 16. CNN Accuracy Chart

In this chart we can see that after 50 epochs how the model was behaving. The final CNN model we can see that there are some fluctuations in validation accuracy on a couple of epochs. One massive dip occurred on the 24th epoch, the accuracy gone down to as low as 59%. Compared to the relatively stable training accuracy.

http://dx.doi.org/10.36596/jcse.v4i2.381

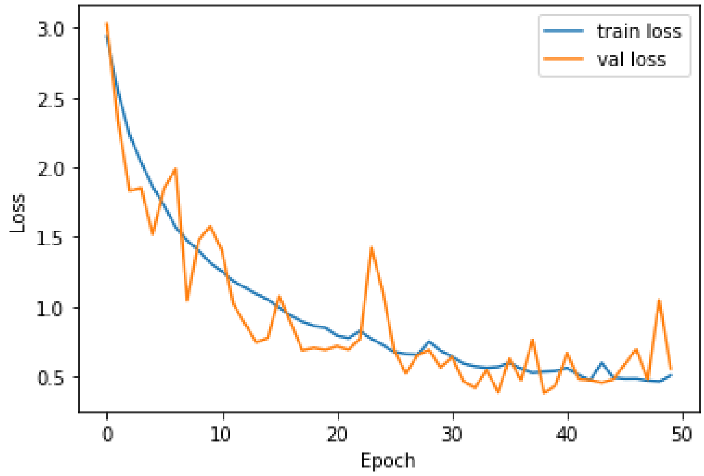

Figure 17. CNN Loss Chart

In this chart we can see that after 50 epochs how the model was behaving. The training loss is projecting a quite stable decrease of loss in 50 epochs with some up ticks here and there. The validation loss also performs quite well but it fluctuates more than the training loss. With a particularly huge spike on the 24th epoch. The loss value reached 1.4. In the end the CNN model achieves: 83.5% on Training Accuracy; 0.5069 on Training Loss 86% on Validation Accuracy; 0.5542 on Validation Loss. This is quite a respectable result for such a simple model design. It is not optimal, but it is good enough for this showcase.

Now, moving on to the ResNet model. Provided below are the Training Accuracy v Validation Accuracy Chart, Training Loss v Validation Loss Chart, End Training Accuracy, End Training Loss, End Validation Accuracy, and End Validation Loss.

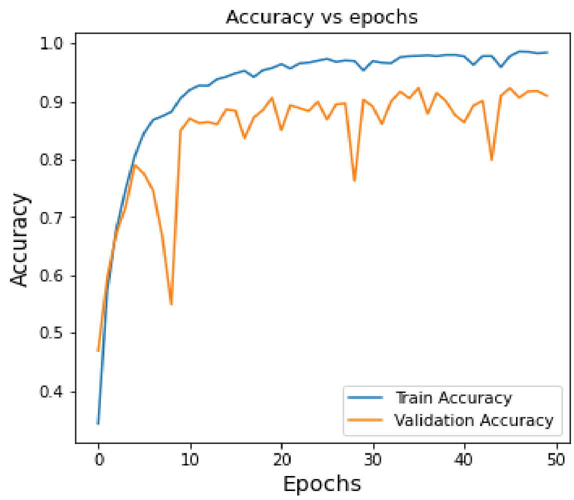

Figure 18. ResNet Accuracy Chart

In this chart we can see that after 50 epochs how the model was behaving. The ResNet model is quite stable during this 50 epoch run. Most notably the training accuracy is high and it does so with tiny fluctuations. While the validation accuracy is also high, almost reaching the training accuracy numbers. While the validation accuracy is quite stable, there are a few dips in accuracy during this run. There is one massive spike on the 8th epoch, the accuracy is going down to only 54%.

http://dx.doi.org/10.36596/jcse.v4i2.381

 e-ISSN 2721-0251

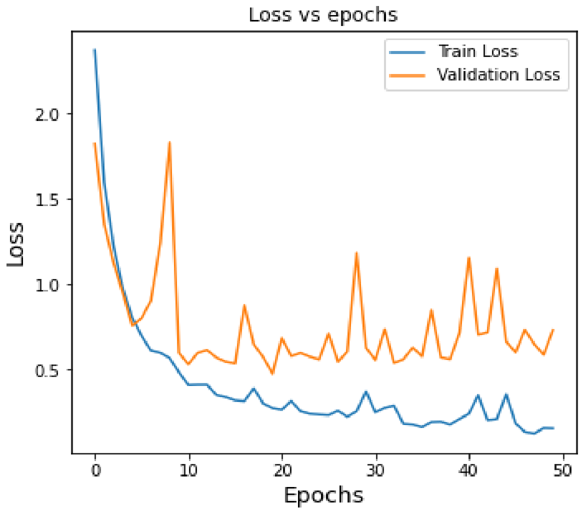

Figure 19. ResNet Loss Chart

In this chart we can see that after 50 epochs how the model was behaving. The ResNet model is quite stable during this 50 epoch run. Most notably the training loss is low and it does so with tiny fluctuations. While the validation accuracy is acceptable and quite stable, there is one massive spike in validation loss on the 8th epoch with the loss value reaching 1.8. In the end the ResNet model achieves: 98.4% on Training Accuracy; 0.156 on Training Loss 90.9% on Validation Accuracy; 0.733 on Validation Loss. The ResNet model managed to outdone the CNN model by quite a margin. Since the ResNet model managed to outdone the CNN model, the categorical distribution shown is only the ResNet categorical distribution.

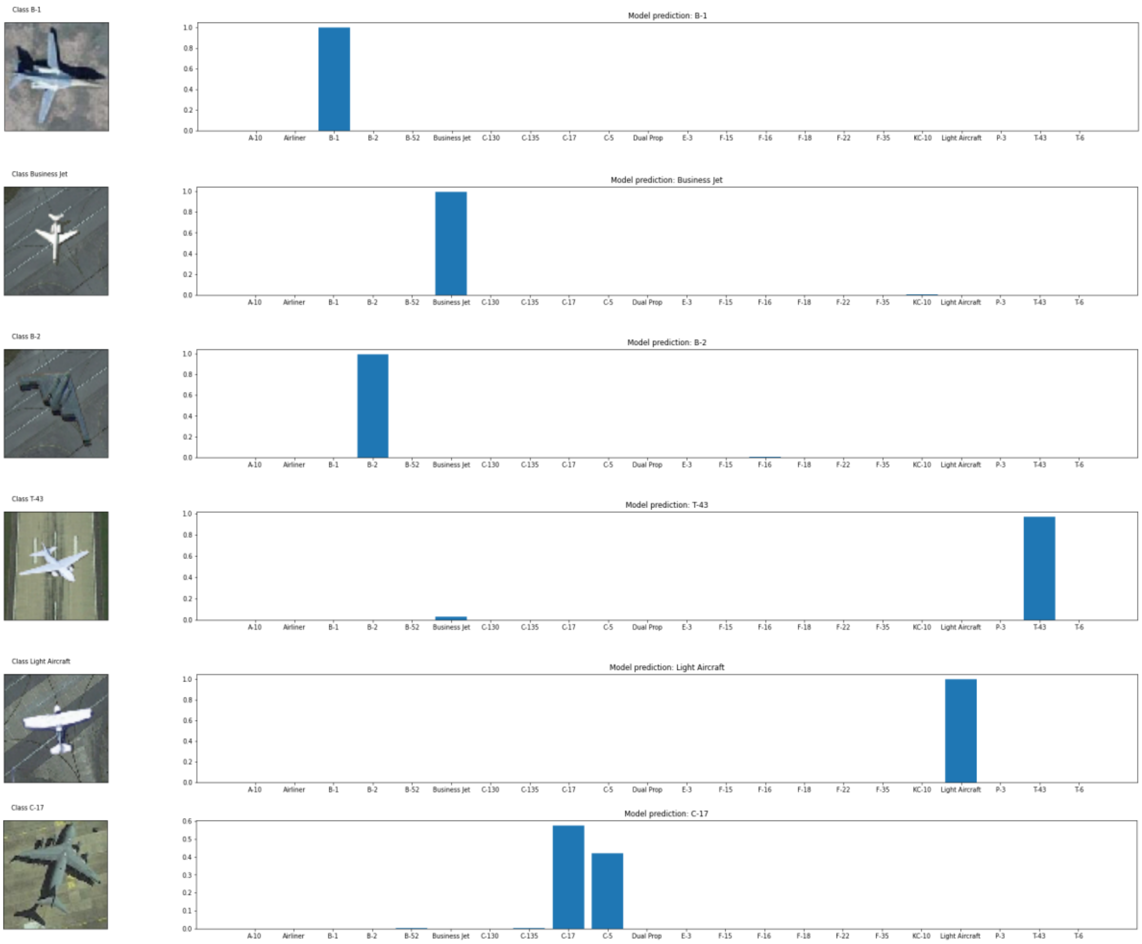

Figure 20. Categorical Distribution 1

http://dx.doi.org/10.36596/jcse.v4i2.381

In figure 20, we can see that the model mistook Business Jet as T43. This missclassing only happened on a small enough scale. What is interesting is that the model mistook C-5, C-135, and B-52 as C-17. It is an expected missclassing because the aircrafts that are mentioned share some similar features. They all are huge aircrafts with four wing mounted engines (8 engines for the B-52 but 2 engines are mounted together per pylon) they all share somewhat similar figures. Granted, the C-5 does share more resemblances with the C-17 if we look at it from the top. So when the model mistook C-5 as C-17 it was not that surprising. But worryingly the mistaken identity happens too frequently for comfort. This skews the overall accuracy of the model.

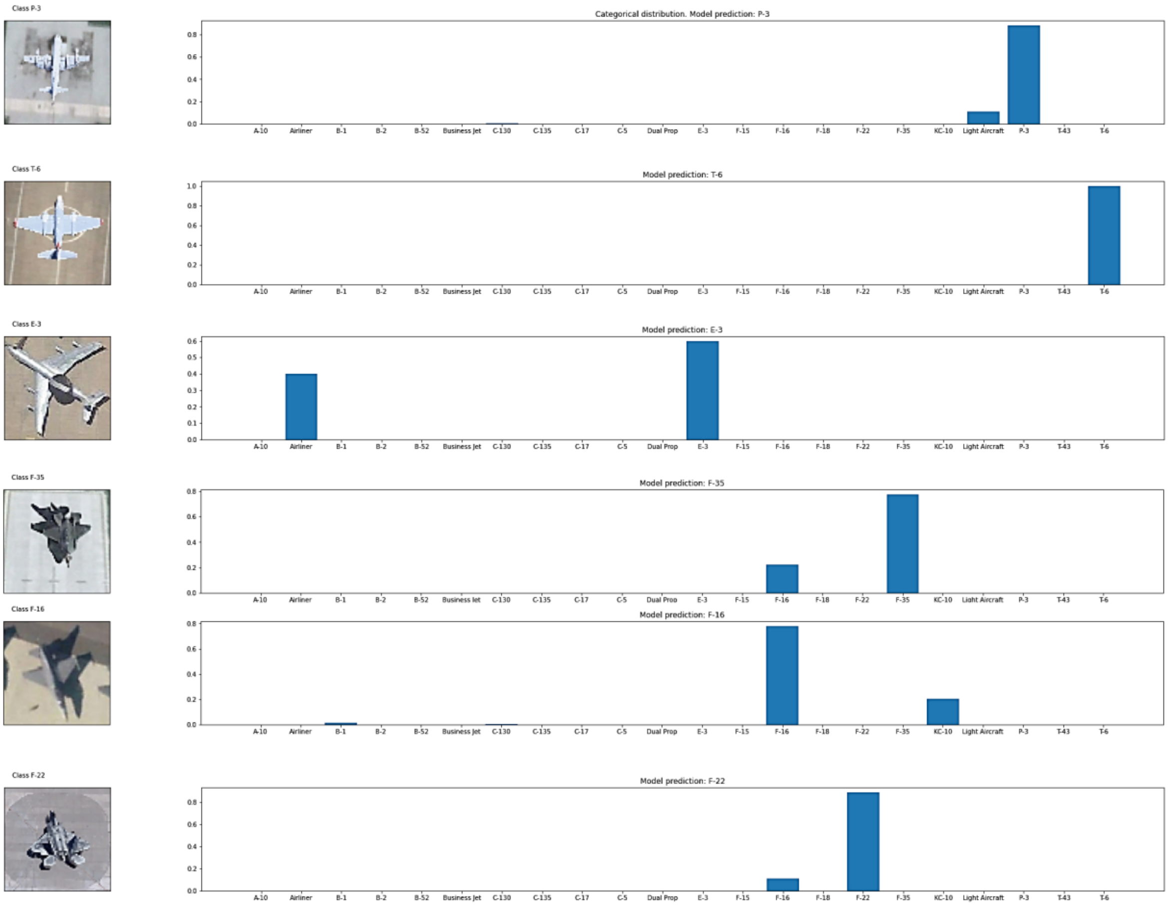

Figure 21. Categorical Distribution 2

The model predicts Light Aircraft as P-3 in some cases. Both of those aircrafts look different from each other. Maybe the model mistook both aircraft's airframes for each other. The model also mistook Airliners as E-3s. The model may have become unable to recognise the shape of the radar on the E3. If we discount that fact, the E3 and Airliners do share a few resemblances although so slightly. Moving on to the F-35s, the model mistook F-16s as F-35s. Again, the model might confuse the aircraft's airframe with each other. Now we have an interesting case of mistaken identity. The model confuses F-16s with KC-10, B-1, and C-130. All four of these aircrafts couldn't be more different from each other. One is a small single engined fighter, one is a huge tanker/cargo aircraft, one is a sleek bomber, and the other is a four engined propeler cargo aircraft. This is a surprising result to say the least. Next up we have F-16s mistaken as F-22s. Same as with the F-35s, the model might confuse the aircraft's airframe with each other.

http://dx.doi.org/10.36596/jcse.v4i2.381

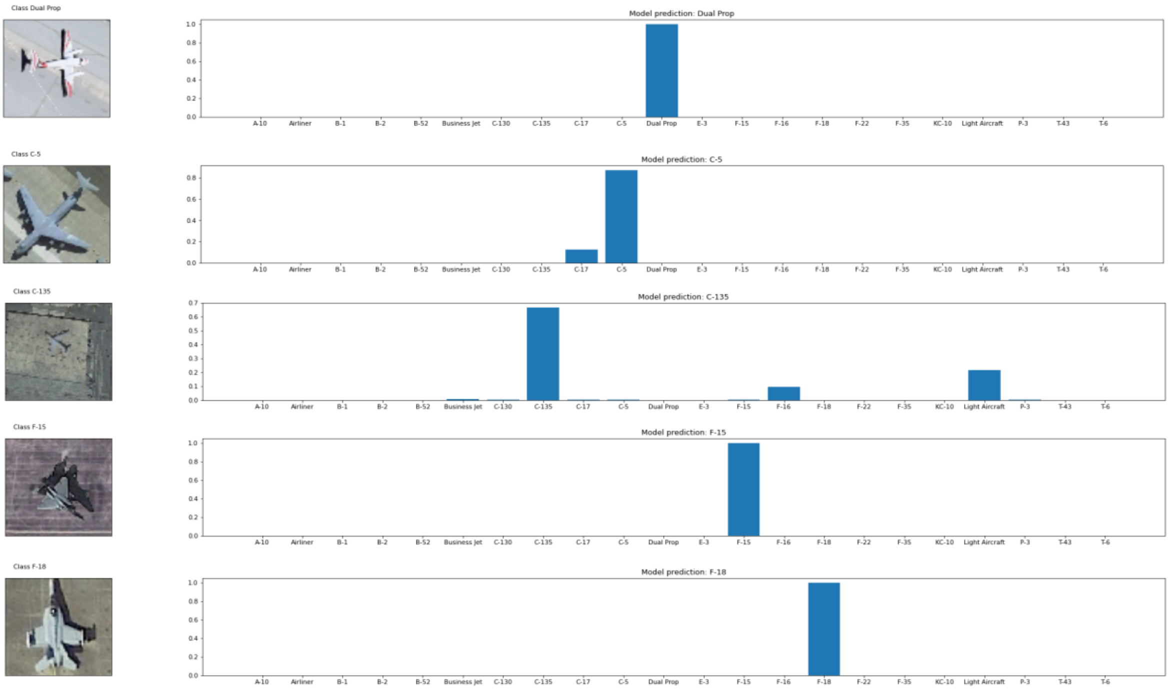

Figure 22. Categorical Distribution 3

As discussed earlier, the confusion between C-5 and C-17 classifications can be somewhat expected due to the shared resemblances, particularly when viewed from the top. Their outlines bear a certain similarity, and while the C-5 appears to have a longer fuselage than the C-17, both exhibit the characteristic of four engines mounted on the wings, as discerned from the images. Thus, the model's occasional misclassification of C-5 as C-17, given these visual cues, is somewhat understandable.

However, venturing into a more intricate aspect, we encounter the perplexing case of the C-135 as illuminated in Figure 22. The C-135 stands out for its remarkable confusion with a diverse array of aircraft, including Business Jet, C-130, C-17, C-5, F-15, F-16, Light Aircraft, and P-3. The model's capacity to mistake the C-135 for such a variety of aircraft is indeed remarkable and introduces a unique challenge. While the comparison with C-5, C-17, and C-130s can be rationalized, considering their shared characteristics as sizable planes with four engines on their wings, the model's tendency to misclassify F-15s, F-16s, Light Aircrafts, P-3s, and Business Jets takes the complexity of classification to an entirely different level.

Figure 22 becomes a visual narrative, highlighting the intricacies and challenges faced by the model in distinguishing between diverse aircraft types within the MTARSI dataset. The misclassifications observed underscore the need for further refinement and training to enhance the model's accuracy and robustness in handling a broad spectrum of aircraft categories.

http://dx.doi.org/10.36596/jcse.v4i2.381

 e-ISSN 2721-0251

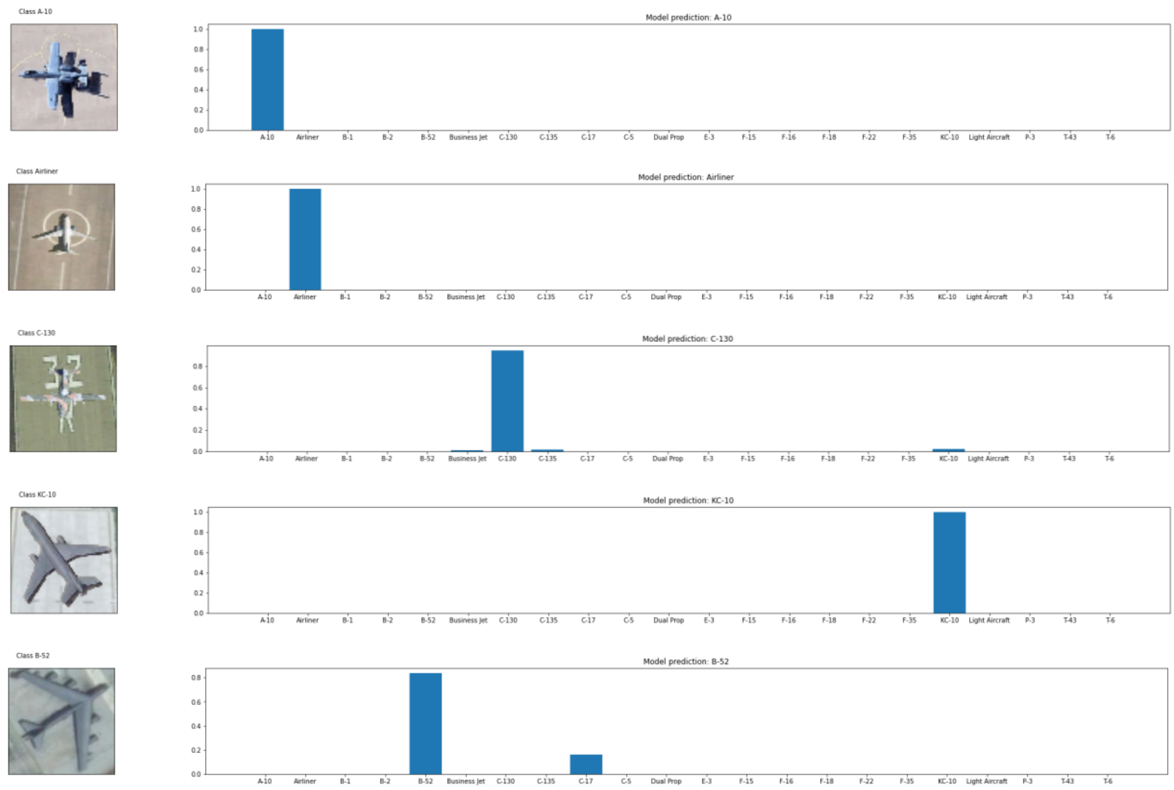

Figure 23. Categorical Distribution 4

The model confused the C-130 with a couple of aircraft. It is confused with Business Jet, C-135, and KC-10. The model confuses Business Jets with C-130 when they do not share any similarities at all. We can say that for C-135 and KC-10 as well. The only thing that the C-130s share with them is their huge size. It is quite interesting that the model confuses the C-130 with those three aircraft. If it were mistaken for P-3s, then it is more expected. Lastly, the model confuses the C-17 with the B-52. Granted, they both are huge planes with four engines jutting out of their wings if we are looking from the top side, this confusion is warranted.

4. Conclusion

Based on the results achieved above, it can be concluded that, in this specific use case with all the parameters mentioned above, ResNet is the more accurate and reliable model compared to CNN. The models above were evaluated using the MTARSI dataset. The dataset cleanliness was questionable at best. We work around this by manually inspecting and relabelling all the mislabelling instances. After that to ensure that our model would not be plagued with overfitting, we augment the dataset using ImageDataGenerator from keras. In the end, the ResNet model managed to edge out the CNN model with a final validation accuracy of 90.9% compared to a validation accuracy of 86% on the CNN model.

References

[1] V. Shah, “Image Processing and its Military Applications”, DSJ, vol. 37, no. 4, pp. 457-468, Jan. 2014. https://core.ac.uk/download/pdf/333721419.pdf

[2] T. Hiippala, “Recognizing military vehicles in social media images using deep learning,” 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), Jul. 2017, doi: 10.1109/isi.2017.8004875.

[3] D. Legendre and J. Vankka, “Military Vehicle Recognition with Different Image Machine Learning Techniques,” Communications in Computer and Information Science, pp. 220–242, 2020, doi: 10.1007/978-3-030-59506-7_19.

http://dx.doi.org/10.36596/jcse.v4i2.381

[4] C. Chen, J. Huang, C. Pan, and X. Yuan, "Military Image Scene Recognition Based on CNN and Semantic Information," 2018 3rd International Conference on Mechanical, Control and Computer Engineering (ICMCCE), Sep. 2018, doi: 10.1109/icmcce.2018.00126.
[5] X. Xiaozhu and H. Cheng, "Object Detection of Armored Vehicles Based on Deep Learning in Battlefield Environment," 2017 4th International Conference on Information Science and Control Engineering (ICISCE), Jul. 2017, doi: 10.1109/icisce.2017.327.
[6] A. Zhao et al., "Aircraft Recognition Based on Landmark Detection in Remote Sensing Images," IEEE Geoscience and Remote Sensing Letters, vol. 14, no. 8, pp. 1413–1417, Aug. 2017, doi: 10.1109/lgrs.2017.2715858.
[7] Qichang Wu, Hao Sun, Xian Sun, Daobing Zhang, Kun Fu, and Hongqi Wang, "Aircraft Recognition in High-Resolution Optical Satellite Remote Sensing Images," IEEE Geoscience and Remote Sensing Letters, vol. 12, no. 1, pp. 112–116, Jan. 2015, doi: 10.1109/lgrs.2014.2328358.
[8] Z.-Z. Wu et al., "A benchmark data set for aircraft type recognition from remote sensing images," Applied Soft Computing, vol. 89, p. 106132, Apr. 2020, doi: 10.1016/j.asoc.2020.106132.
[9] Xu Y, Jia R, Mou L, Li G, Chen Y, Lu Y, Jin Z. Improved relation classification by deep recurrent neural networks with data augmentation. 2016 Jan 14. arXiv preprint arXiv:1601.03651.
[10] Wang J, Perez L. The effectiveness of data augmentation in image classification using deep learning. Convolutional Neural Networks Vis. Recognit. 2017 Dec;11:1-8.
[11] Asperti A, Mastronardo C. The effectiveness of data augmentation for detection of gastrointestinal diseases from endoscopical images.2017 Dec 11. arXiv preprint arXiv:1712.03689.
[12] M. V. Valueva, N. N. Nagornov, P. A. Lyakhov, G. V. Valuev, and N. I. Chervyakov, "Application of the residue number system to reduce hardware costs of the convolutional neural network implementation," Mathematics and Computers in Simulation, vol. 177, pp. 232–243, Nov. 2020, doi: 10.1016/j.matcom.2020.04.031.
[13] Liu S, Liu Z. Multi-channel CNN-based object detection for enhanced situation awareness. 2017 Nov 30. arXiv preprint arXiv:1712.00075.
[14] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian. (2016). Deep Residual Learning for Image Recognition. 770-778. 10.1109/CVPR.2016.90.
[15] He, F., Liu, T., & Tao, D. (2020). Why ResNet Works? Residuals Generalize. IEEE Transactions on Neural Networks and Learning Systems, 1–14. doi: 10.1109/tnnls.2020.2966319

http://dx.doi.org/10.36596/jcse.v4i2.381