APS360 PROJECT FINAL REPORT: COMMERCIAL AIRCRAFT MODEL CLASSIFICATION

Joey Yizhi Li
Student# 1010877044
joeyyizhi.li@mail.utoronto.ca

Ziyu Chen
Student# 1009857488
ziyujacob.chen@mail.utoronto.ca

Jici Ye
Student# 1009403879
jici.ye@mail.utoronto.ca

Deren Zhang
Student# 1008828440
deren.zhang@mail.utoronto.ca

1 INTRODUCTION

In this project, we propose a deep learning-based image classification system capable of recognizing commercial aircraft models from photographs. Accurately identifying commercial aircraft models from images has valuable applications in air traffic management, aviation analytics, and defense surveillance. However, manual identification of aircraft models from images is a demanding task, typically requiring extensive domain expertise. Differentiating between models often hinges on nuanced features such as slight differences in engine nacelle placement, wing sweep angle, landing gear configuration, or tail structure— details that are easily overlooked by the untrained eye. Thus, a machine learning model is the most effective approach to this problem.

Figure 1: Boeing 737-500, adapted from Maji et al. (2013c)

We build upon the FGVC-Aircraft dataset Maji et al. (2013a) by incorporating additional, up-to-date images of modern aircraft models and performing extensive data cleaning to ensure high-quality inputs. A sample input is shown in Figure 1, and we want the model to output its corresponding variant type. We choose a traditional Bag-of-Visual-Words (BoVW) model with a Support Vector Classifier (SVC) and an ANN classifier as the baseline. Our main model explores state-of-the-art convolutional neural network (CNN) architectures, namely VGG-19 Simonyan & Zisserman (2015), ResNet-50 He et al. (2015), and DenseNet-121 Huang et al. (2018), all modified with a Spatial Pyramid Pooling (SPP) layer to accommodate images of varying sizes without the need for explicit resizing. This work aims to assess the effectiveness of modern CNNs against classical methods in fine-grained, multi-class aircraft classification and to identify model architectures that generalize well under diverse viewing conditions. Our results will inform future efforts in applying AI for aviation-related visual recognition tasks.

2 Background and Related Work

One of the earliest large-scale datasets for aircraft model classification, FGVC-Aircraft Maji et al. (2013a), provided over 10,000 labeled images across 100 aircraft variants. This dataset laid the foundation for fine-grained classification model, and was used in the 2013 ImageNet FGVC challenge. It benchmarked with the traditional Support Vector Classifier (SVC) with Bag-of-Visual-Words (BoVW) and SIFT approach, and obtained a poor accuracy of 48.68%.

Image classification has seen significant advances in recent years, primarily driven by deep convolutional neural networks trained on large-scale datasets such as ImageNet. Several state-of-the-art architectures have emerged from this progress, including AlexNet Krizhevsky et al. (2012), VGGNet Simonyan & Zisserman (2015), ResNet He et al. (2015), EfficientNet Tan & Le (2019), and Vision Transformers (ViT) Dosovitskiy et al. (2021). These models differ in depth, width, parameter efficiency, and architectural innovations, such as the 3×3 kernel in VGG, residual connections in ResNet, and squeeze-and-excitation blocks in EfficientNet.

2.1 Aircraft Nomenclature

We can classify aircraft through three different levels: manufacturer, model (family), or variant. The manufacturer is the company that designs and produces the aircraft. A model refers to a general aircraft design with shared characteristics (such as A350 and B737). Variants refer to the sub-models with slight modifications (e.g., fuselage length, engine type, range). For example, the most popular and long-lasting B737 family has over 23 variants. Variants under the same model can be very similar and hard to classify. Differences may be internal (avionics, engines) or minor structural changes (stretched fuselage).

3 Data Processing

3.1 Data Collection

We build upon the FGCV-Aircraft dataset Maji et al. (2013c), which includes 100 aircraft variants, each with 100 images. Since it was created in 2013, it does not include any latest aircraft models. Additionally, it contains some military aircraft, such as F-16. Thus, we made a final list of 59 variants of only commercial aircraft to be included in our final dataset by incorporating 15 new variants and 45 variants from the original dataset:

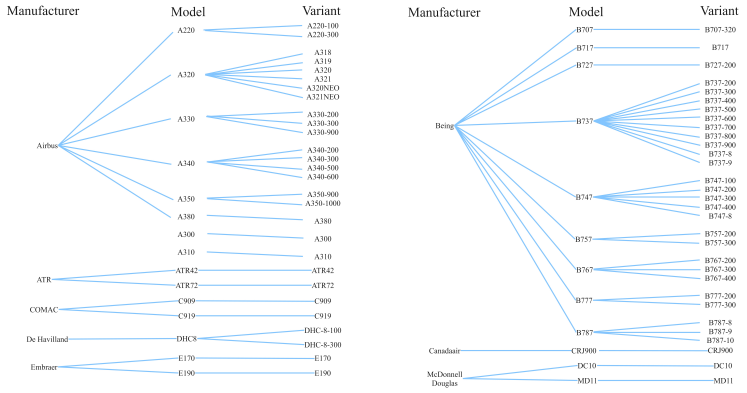


Figure 2: This graph shows the final list of airplane models that we decided to include in our final dataset.

For the 15 newly-added variants, we manually collected 100 images for each of them. For image collections, we follow the following criteria:

1. Different Airline Liveries
2. Different Camera

3. View of the full aircraft body

According to these rules, when collecting images for best-selling models such as B737-8, A320NEO, A321NEO, etc, we mainly focused on choosing different liveries and angles for them. However, for A220-100, B747-8, and other low-sales models, we mainly focused on choosing images that have different angles, environments, and lights. After choosing suitable images, we downloaded them into their sets and named them from 1-100 to keep track of the number of images that have already been collected. All images are in the form of *.jpg* and they are uploaded to the shared Google Drive folder for further actions.

As the names of the variants use different suffixes to indicate different seat layouts, engine options, etc., we used the following search names:

Aircraft Name	Search Name in JetPhotos
A220-100	Airbus A220-171
A220-300	Airbus A220-371
A320NEO	Airbus A320-251/271N
A321NEO	Airbus A321-251/271NX
A330-900	Airbus A330-941
A350-900	Airbus A350-941
A350-1000	Airbus A350-1041
B737-8	Boeing 737-8 Max
B737-9	Boeing 737-9 Max
B747-8	Boeing 747-89L/830/8B5
B787-8	Boeing 787-8 Dreamliner
B787-9	Boeing 787-9 Dreamliner
B787-10	Boeing 787-10 Dreamliner
C909	COMAC C909/ARJ21-700
C919	COMAC C919

Figure 3: The figure shows the list of names that were used to search the specific aircraft in Jetphotos (2002). The difference between the search name and the Aircraft name for Airbus products comes from the engine selection. For A320NEO and A321NEO, '-251' represents the PW engine and '-271' represents the CFM engine. For Boeing 747-8, the difference comes from the suffix named by the operator. They represent Air China, Lufthansa, and Korean Air, respectively. For COMAC C909, ARJ21-700 and C909 refer to the same aircraft as COMAC officially renamed the aircraft on November 12th, 2024, at the China International Aviation & Aerospace Exhibition.

At the end, we randomly partitioned the dataset into a 70-15-15 train-val-test split.

3.2 Data Inspection & Cleaning

```
Checking: 220-100
Checking: 220-300
Checking: 330-900
Duplicate: /content/drive/MyDrive/Colab Notebooks/Newly added images/330-900/23.jpg is a copy of /content/drive/MyDrive/Colab Notebooks/Newly added images/330-900/6.jpg
Checking: 350-900
Checking: 737-9
Checking: 747-8
Checking: 350-1000
Checking: 787-8
```

Figure 4: Output of the automated script showing that a duplicate has been found in A330-900 on pictures 6 and 23

After completing 15 classes of airplane model collections with 100 images per category, we implemented automated inspection scripts to detect corrupted or unreadable images and to identify duplicate files. Duplicate images were found in the A330-900 model category, as shown in Fig. 4, and were manually removed and a new image was added instead.

As we used Spatial Pyramid Pooling (SPP), it eliminates the need for image resizing or enforcing uniform resolution. However, our criteria indicate that each image should capture the entire airplane, as many aircraft models have similar features when only parts (such as engines or wings) are visible, and partial views can be misleading. To ensure this, all images were manually reviewed, and those

showing only partial aircraft were excluded. We then manually add new images to replace the removed images so that each aircraft model would have 100 images.

However, the photos we downloaded from *JetPhotos.com* Jetphotos (2002) contain a 20 pixel frame and watermark on the photos, so we wrote a code to delete the bottom 20 pixels and manually removed the watermark in the photos.

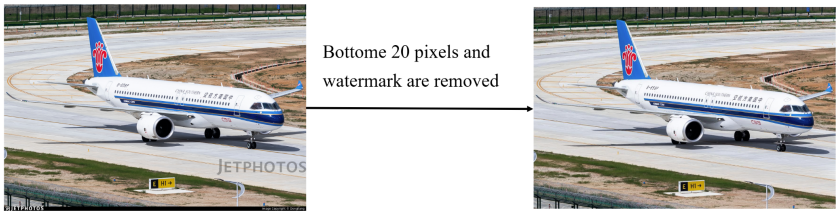


Figure 5: This photo shows an example of the original downloaded photo and the photo that after the complete cleaning process.

3.3 Data Augmentation

As the aircraft dataset contains a large amount of variants in each class, the team noticed a significant issue with overfitting. To mitigate this, regularization methods such as early stopping and the ADAM optimizer were applied during training. Data augmentation was also essential, as the model needed to learn the underlying features of each airplane model rather than memorizing them.

Data augmentation was implemented through code after the dataset was finalized, and it was confirmed that all photos are clean. After considering the number of samples and runtime, we decided to perform 7 augmentations on the training dataset, which increased its size by a factor of 8. The augmentations are:

1. Horizontal Flip
2. Vertical Flip
3. Greyscale Conversion
4. Rotation 90°, 180°, 270°
5. Gaussian Blur

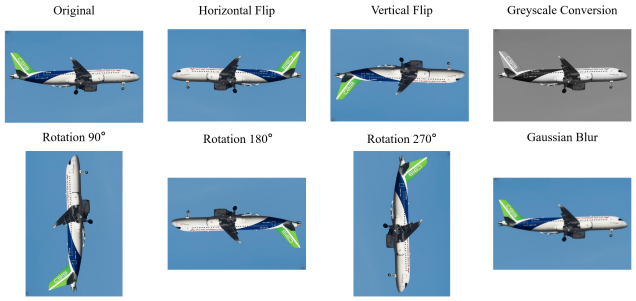


Figure 6: This photo shows an example of the original photo(after data cleaning process) and 7 samples of different augmentations.

4 Model Architecture

Many image classification models exist, most of which rely on convolutional neural networks (CNNs). A typical CNN model consists of two parts: convolutional layers and fully connected (FC) layers. While convolutional layers can accept inputs of varying dimensions, FC layers require a fixed-length input vector. However, images in the dataset often vary widely in size. Although

resizing techniques such as cropping or warping can be applied, they involve excessive manual effort and may distort image structures, potentially degrading the CNN's ability to accurately extract features.

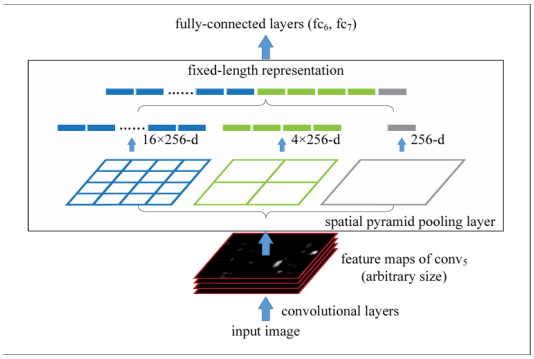


Figure 7: Spatial Pyramid Pooling Layer. 1×1, 2×2, and 4×4 regions are used. 256 feature maps are converted into a fixed-length 5376-dimensional vector. Adapted from He et al. (2014).

To address this issue, we implement a *Spatial Pyramid Pooling* (SPP) layer at the end of the convolutional layers. Originally proposed by He et al. (2014), SPP generates a fixed-length representation regardless of input image size. Specifically, the feature map produced by the convolutional layers is partitioned into increasingly coarser spatial bins (e.g., 1×1, 2×2, 4×4 regions), and within each bin, a pooling operation (typically max or average pooling) is applied. Regardless of the original feature map dimensions, each pooling region produces a single value. By concatenating these pooled values across all bins and levels, a fixed-length feature vector is obtained, which can be passed to the FC layers without requiring the input images to have uniform dimensions.

To find the best convolutional backbones, we implement the pretrained **VGG-19** Simonyan & Zisserman (2015), **ResNet-50** He et al. (2015), and **DenseNet-121** Huang et al. (2018), and replace their last pooling layer with a maximum SPP layer with spatial bins 1×1, 2×2, and 4×4. After extracting the features of all images, we then train a two-layer ANN classifier to predict the variant of the images.

5 BASELINE

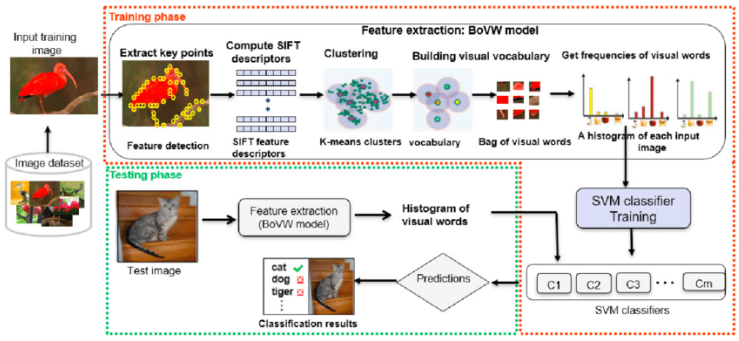


Figure 8: An Illustration of the BoVW+SVC Approach in Image Classification Task. Adapted from Filali & Zghal (2021)

The authors of the dataset used a Support Vector Classifier (SVC) model as a baseline Maji et al. (2013b). Specifically, they selected a non-linear SVM on a χ^2 kernel, bag-of-visual words (BoVW), 600 k-means words dictionary, multi-scale dense SIFT features, and 1×1, 2×2 spatial pyramid. Essentially, the model first extracts local texture patterns like the number of engines, the shape of wings, and more; then counts their occurrence (as visual words); notes their rough spatial location via a spatial pyramid; and lastly uses an SVM to learn patterns in these histograms to separate

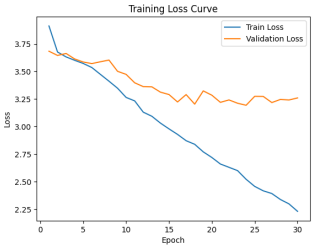
airplane classes. An example pipeline is shown in Fig. 8. Since this is the baseline the authors of the dataset used Maji et al. (2013c), we chose this approach as our baseline model. Due to the colab memory constraint, the K-means model is constructed by randomly selecting 150 images from the unaugmented testing set.

However, since this paper was published in 2013, this traditional approach does not take advantage of the advances and power of deep learning networks. To improve performance, we modify the pipeline by replacing the SVM classifier with a three-layer MLP in the final classification step.

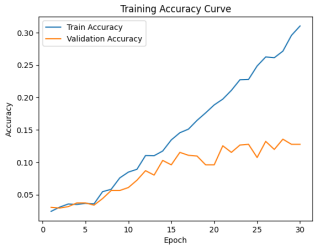
6 Results

6.1 Quantitative Result

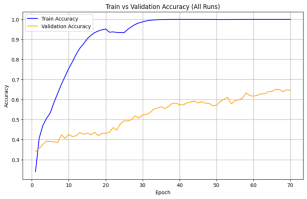
6.1.1 Accuracy and Loss



(a) BoVW+MLP Loss over Epoch



(b) BoVW+MLP Accuracy over Epoch



(c) ResNet+SPP Accuracy

Figure 9: Training performance comparison between BoVW+MLP and ResNet+SPP models

Table 1: Hyperparameter Choice and Accuracy and Loss Curve of the SPP Models and Baseline Models

	VGG+SPP	ResNet+SPP	DenseNet+SPP	BoVW+SVC	BoVW+MLP
Learning Rate	1e-3+3e-4	1e-3+3e-4	8e-3	N/A	9e-5
Batch Size	32	32	32	N/A	64
Number of Epoch	20+40	20+25+25	20	N/A	30
Training Accuracy	100%	100%	96.36%	17.4%	31.01%
Training Loss	0.7637	0.2844	0.7521	N/A	2.2301
Validation Accuracy	62.71%	65.08%	62.57%	N/A	14.92%
Validation Loss	2.1570	1.6895	2.2966	N/A	3.2600

We used CrossEntropyLoss for all 59 types of variants to calculate the loss during training. The Adam optimizer is chosen to train the model. The hyperparameters during training, including learning rate, batch size, and number of epochs, are decided through optimizing the model performance. Since SVC training does not need a validation set, the validation accuracy and loss are omitted.

Figures 9a, 9b, and 9c illustrate the training performance of the BoVW+MLP and ResNet+SPP models, respectively. For the BoVW+MLP model, the training loss consistently decreases over the epochs, while the validation loss shows slower improvement, indicating potential underfitting. In contrast, the ResNet+SPP model exhibits a rapid increase in both training and validation accuracy during the initial epochs, with the validation accuracy stabilizing afterward. The validation accuracy remains consistently lower than the training accuracy, suggesting a moderate degree of overfitting.

Table 1 summarizes the chosen hyperparameters, as well as the final training and validation results for all evaluated models. ResNet+SPP achieves the lowest training loss (0.2844) and the highest validation accuracy (65.08%), outperforming the VGG+SPP (62.71%) and DenseNet+SPP (62.57%) models. Both baseline models perform badly. BoVW+MLP performs significantly worse, with only 14.92% validation accuracy, despite achieving a higher training accuracy of 31.01% compared to BoVW+SVC (17.4%).

The results confirm that SPP-based deep learning architectures, particularly ResNet+SPP, offer superior generalization performance compared to both traditional BoVW-based methods and other tested deep learning backbones. We select the ResNet+SPP model as our final model.

6.1.2 Confusion Matrix

We can measure the accuracy of the model through three levels: variants, family, and manufacturers (although the models are trained solely on variant classification). For the baseline BoVW+MLP model and ResNet+SPP model, we calculated the corresponding validation confusion matrix for each classification level.

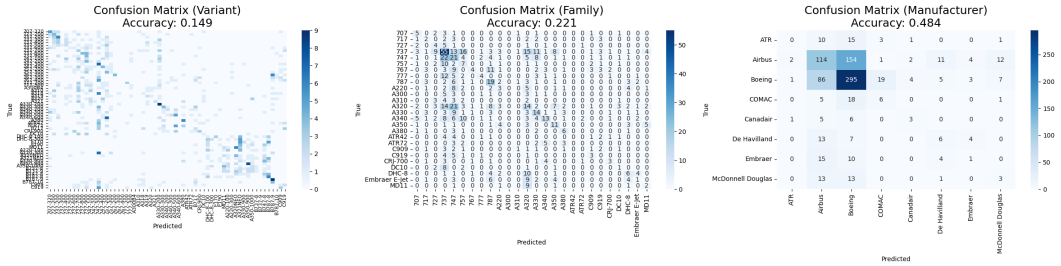


Figure 10: BoVW+MLP Validation Confusion Matrices

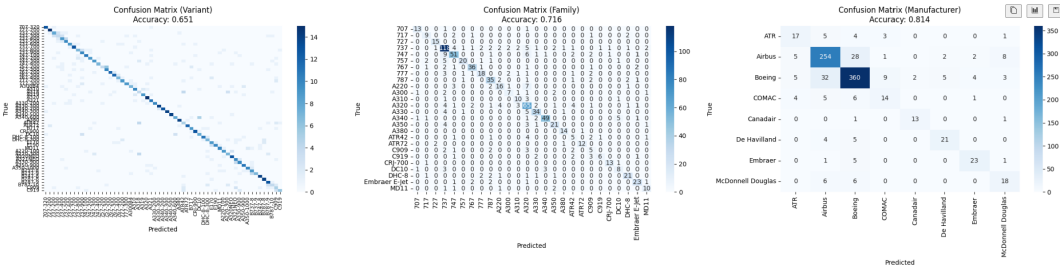


Figure 11: ResNet+SPP Validation Confusion Matrices

The confusion matrices (Figures 10 and 11) show that ResNet+SPP consistently outperforms BoVW+MLP across all classification levels. At the variant level, ResNet+SPP achieves 65.08% accuracy versus 14.92% for BoVW+MLP, with far fewer misclassifications and more correct predictions over the diagonal. Accuracy further improves to 84.11% at the family level and 97.63% at the manufacturer level, compared to 42.03% and 84.81% for BoVW+MLP. These results highlight the superior fine-grained recognition capability of ResNet+SPP over the traditional BoVW approach.

6.2 Qualitative Result

We analyze common prediction scenarios of the model by randomly selecting images from the validation set, and three representative examples are shown in Figure 12. The left image illustrates a correct prediction of a B747-400 with high confidence, likely due to the aircraft's distinctive double-deck design and the clear background. The middle image shows a correct prediction of an A350-900 with low confidence, where a cluttered background may have made the classification more challenging. The right image depicts an incorrect prediction of another variant within the same family, which is expected because variants in the same family share highly similar visual features.

7 Performance on New Data

We applied the ResNet+SPP model on the test set and obtained the confusion matrix shown in Figure 13. The test performance is consistent with the validation performance in Figure 11, demonstrating that the model generalizes well without significant overfitting. It achieves accuracies of 64.7%,

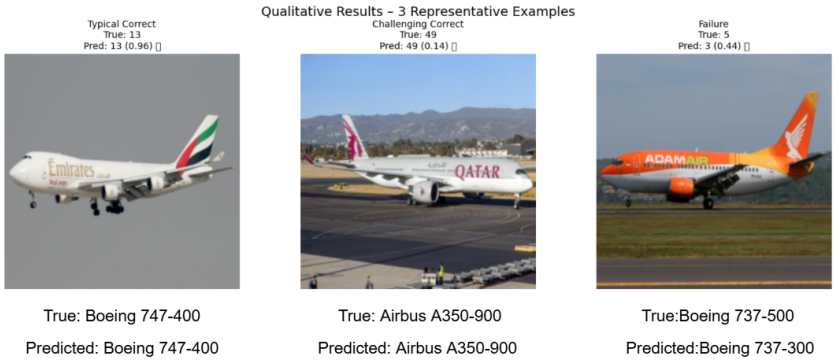


Figure 12: Qualitative Result of the ResNet+SPP Model

73.6%, and 81.9% for variant, model, and manufacturer classification, respectively. The majority of predictions are concentrated along the diagonal, indicating that most classes are correctly classified. Similarly, a lot of misclassification occurs in the family B737, where many variants are classified as another B737 variant. These examples highlight that the model performs well when distinctive features are visible and the background is clean, but struggles with subtle inter-variant differences or visually complex scenes.

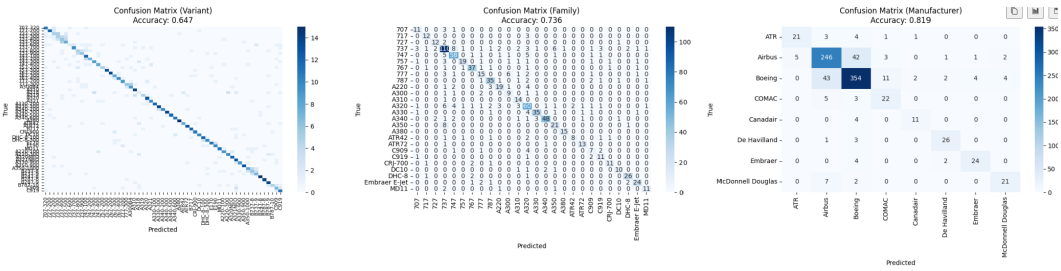


Figure 13: ResNet+SPP Testing Confusion Matrix

To provide a meaningful reference point, one of the team members—an expert in commercial aircraft models—manually identified the aircraft variants in the testing set. The resulting performance, shown in Figure 14, serves as a human baseline. This expert achieved accuracies of 83.39%, 96.16%, and 98.42% in variant, model, and manufacturer classification, respectively, substantially surpassing the model's performance.

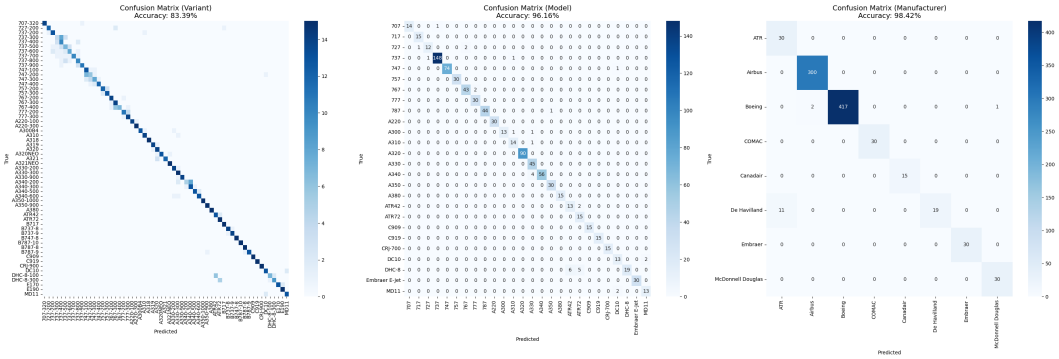


Figure 14: Human Baseline Testing Confusion Matrix

8 DISCUSSION

Our experiments demonstrate that the ResNet+SPP model effectively classifies commercial aircraft images across variant, model, and manufacturer levels. The confusion matrix reveals strong performance, with most predictions concentrated along the diagonal. Misclassifications primarily occur between visually similar variants of the same model, a finding supported by qualitative analysis. These errors highlight the inherent challenges of fine-grained classification, particularly when distinguishing variants with nearly identical visual features. Notably, the model also frequently confuses Boeing and Airbus aircraft, likely due to their shared characteristics as large commercial jet manufacturers—a phenomenon previously reported by Maji et al. (2013c).

Comparison with a human baseline further underscores the model's limitations. While it captures broad visual patterns, it struggles with subtle distinctions requiring domain expertise. This performance gap is substantial but not unique to our model: even human evaluators found certain variants challenging to distinguish. For instance, our baseline annotator reported difficulty identifying B737 variants, as reflected in Figure 14.

Overall, our findings indicate that the ResNet+SPP architecture is robust for general aircraft classification, but additional strategies—such as incorporating domain-specific features or expanding the training dataset—may be necessary to approach human-level performance, particularly for fine-grained distinctions. These insights provide a foundation for future improvements in automated aircraft recognition systems.

9 ETHICAL CONSIDERATIONS

This project is intended for educational and research purposes only. Several ethical aspects are carefully considered to ensure the responsible development and deployment of the system:

- **Use of Publicly Available Data:** All images used are sourced from publicly available datasets or open-access platforms. We do not use any proprietary, classified, or personally sensitive material. The focus remains on publicly released commercial aircraft, avoiding any risk of unauthorized data usage.
- **Privacy and Surveillance:** Although the technology has potential defense or surveillance applications, we explicitly limit our scope to aircraft recognition in non-sensitive, civilian contexts. No efforts are made to analyze content beyond aircraft models, and the model is not trained on images containing identifiable individuals, private property, or restricted airspaces.
- **Bias and Representation:** We strive to build a balanced and diverse dataset that includes various commercial aircraft models, lighting conditions, and angles to avoid model bias toward specific aircraft types or operational environments.

10 PROJECT DIFFICULTY

We believe this project exceeds the scope of the lab requirements and academic syllabus for the following reasons:

- Rather than relying on traditional image resizing techniques to handle inputs of varying dimensions, we introduce a novel integration of a Spatial Pyramid Pooling (SPP) layer into a pretrained convolutional backbone. This allows the model to accept inputs of arbitrary sizes without distortion.
- We benchmark the model's performance against a human baseline. Although the model does not surpass human accuracy, establishing this comparison involved significant manual labeling effort and highlights the complexity of the task.
- We experiment with multiple convolutional backbones, including ResNet-50, VGG-19, and DenseNet-121, each modified with the SPP layer. We conduct comparative evaluations and select the architecture that achieves the best performance under our evaluation metrics.

REFERENCES

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*, 2021.

Jalila Filali and Hajer Zghal. Comparing hmax and bovw models for large-scale image classification. *Procedia Computer Science*, 192:1141–1151, 01 2021. doi: 10.1016/j.procs.2021.08.117.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (eds.), *Computer Vision – ECCV 2014*, pp. 346–361, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10578-9.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015. URL https://arxiv.org/abs/1512.03385.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018. URL https://arxiv.org/abs/1608.06993.

Jetphotos. Aviation photos - 5 million+ on jetphotos, 11 2002. URL https://www.jetphotos.com/.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60:84–90, 05 2012. URL https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013a.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, 2013b. URL https://arxiv.org/abs/1306.5151.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, 06 2013c.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015. URL https://arxiv.org/abs/1409.1556.

Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *CoRR*, abs/1905.11946, 2019. URL http://arxiv.org/abs/1905.11946.