# APS360 PROJECT PROPOSAL: 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

## ABSTRACT

Accurately identifying commercial aircraft models from images has valuable applications in air traffic management, aviation analytics, and defense surveillance. In this project, we propose a deep learning-based image classification system capable of recognizing commercial aircraft models from photographs. 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. Our approach explores state-of-the-art convolutional neural network (CNN) architectures, namely VGG-19 and ResNet-50, both modified with a Spatial Pyramid Pooling (SPP) layer to accommodate images of varying sizes without the need for explicit resizing. To benchmark our results, we choose a traditional Bag-of-Visual-Words (BoVW) model with a Support Vector Classifier (SVC) as the baseline. 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.

**Keywords:** Aircraft classification, CNN, Spatial Pyramid Pooling, image recognition, SVC, BoVW

## 1 INTRODUCTION

The aviation industry spans thousands of aircraft models, each with unique structural characteristics including engine configuration, wing design, fuselage shape, and dimensional specifications. Accurate aircraft model identification holds critical application value in air traffic control monitoring, defense reconnaissance, aviation maintenance, historical archives, and enthusiast communities. 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.

This project aims to develop a high-precision aircraft model recognition system based on deep learning. The task presents considerable technical challenges, requiring the system to overcome variable factors such as lighting conditions, viewing angles, and image clarity to accurately capture subtle feature differences in engine nacelle positioning, wing sweep angles, landing gear configuration, and other distinguishing characteristics. Older rule-based image processing methods struggle with such complex scenarios, whereas deep learning technologies, particularly convolutional neural networks (CNNs), have demonstrated exceptional performance in fine-grained image recognition due to their powerful feature extraction capabilities.

By training models on extensively annotated aircraft image datasets, we expect not only to achieve high-accuracy multi-model classification but also to build a comprehensive recognition system adaptable to diverse conditions. This research brings substantial practical value, providing intelligent support for aviation safety monitoring, aircraft manufacturing, and maintenance operations. Ultimately, the project will drive the advancement of intelligent development in the aviation sector.

## 2 Background and Related Work

Automated aircraft recognition from images has been a long-standing challenge in both military and civilian applications, involving tasks like surveillance, traffic monitoring, and aircraft cataloging. Recent advances in computer vision, particularly using deep learning, have significantly improved the feasibility and accuracy of aircraft classification systems.

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

.

Cheng et al. (2016) explored the use of deep CNNs combined with multi-scale feature extraction for aerial image classification. Their approach demonstrated that features extracted at different spatial resolutions improved aircraft recognition performance, especially for small and overlapping targets.

Another key contribution came from Liu et al. (2018), who applied a hybrid CNN + attention mechanism model to aircraft recognition. The attention module allowed the network to focus on discriminative parts of the aircraft, such as engine position or tail shape, which are crucial for distinguishing between visually similar models.

In recent years, transformer-based models such as Vision Transformer (ViT) and Swin Transformer have also been adapted for fine-grained object recognition, including aircraft. Dosovitskiy et al. (2021) showed that ViT can outperform CNNs when trained on large datasets, providing better context modeling across the entire image.

Finally, synthetic data generation techniques using 3D aircraft models (e.g., Su et al. (2015)) have been used to augment training data under varied lighting and angles. These methods address the scarcity of labeled aircraft images and improve generalization to real-world conditions.

These prior works demonstrate the effectiveness of deep learning approaches, especially CNNs and transformers, in tackling the complex problem of aircraft model recognition. Our project builds upon these foundations to further explore lightweight, accurate classification of aircraft using deep neural networks trained on real-world image data.

## 3 Data Processing

We found a dataset that is suitable for our training goals: FGVC-Aircraft dataset Maji et al. (2013a). The images from this dataset were originally sourced from the website *Airlines.net* Airliners (1995), a community-driven aircraft photo-sharing website. However, the dataset is not up to date as it lacks most of the recent commercial airplane models such as Boeing 787-8/9/10, Airbus 350-900/1000 etc. Since our goal is to develop a model that is capable of recognizing the aircraft currently being used, we will add new groups and images to ensure the dataset has broader coverage.

To build our new dataset, we will begin by making a list of commercial airplane models that are missing from the original dataset. The following two pictures will show the airplane models with their manufacturer, family, and variants in the Figures 1 and 2:

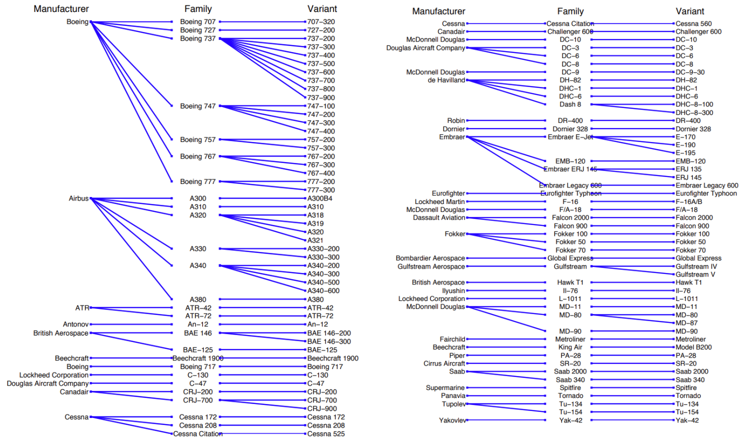


Figure 1: This graph shows the list of airplane models that already existed in the dataset. It includes the manufacturer, family, and variant of these aircraft Maji et al. (2013b). There are 100 commercial and military aircraft models, each with 100 photos.

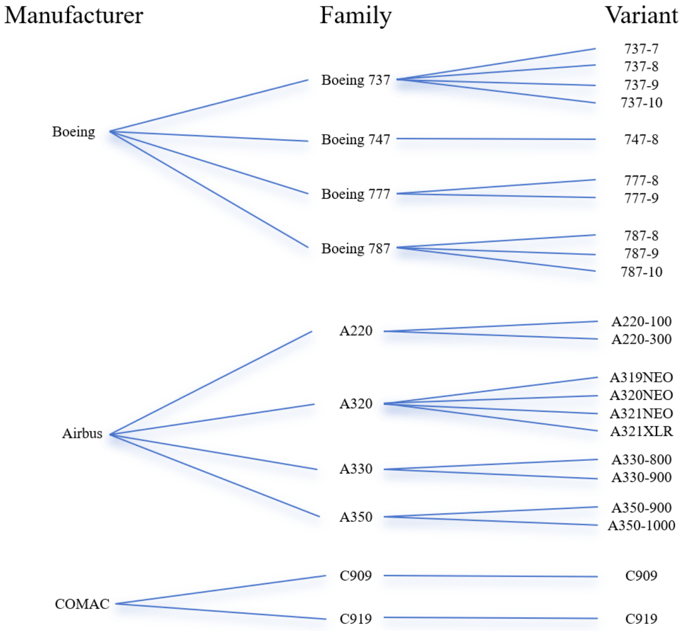


Figure 2: This graph shows the list of airplane models that are missing from the dataset. We will manually add images to update the dataset.

We will then use *Airplanes.net* Airliners (1995) and *JetPhotos.com* Jetphotos (2002) to collect new photos. For each aircraft model, we will gather photos by using the *Flightradar24* to find the aircraft that are currently flying and exist in the real-world ATC traffic radar. We will gather the images with the following criteria:

1. Different Airline Liveries
2. Different Camera
3. View of the full aircraft body

Next, we will conduct the data inspection and exploration process:

1. Ensuring consistent image formats and resolutions (for resolution, we do not want all of our photos to be quite explicit because under some circumstances, the photos of the jetplanes are taken from a considerable distance, we hope the model can successfully identify the planes in those blurry images.)
2. Removing any corrupted or unusable file - This includes images that are corrupted which means they cannot be opened and the images that are not useful for our training. We will use automated scripts to detect unreadable files and manually review the files with issues to ensure the quality of the dataset.
3. Identifying and Removing duplicates - We will treat two photos that are in the same angle or same livery as repeated images. We hope our model can ignore the effect of different liveries on the aircraft.)
4. Excluding images that show only partial aircraft - Our model is designed to identify the planes as a whole body, which means images of partial aircraft, such as engines, wings, and gears, etc. The reason for this is that some planes are very similar at some parts, which can cause a real high difficulty to identify even for an aviation aficionado.

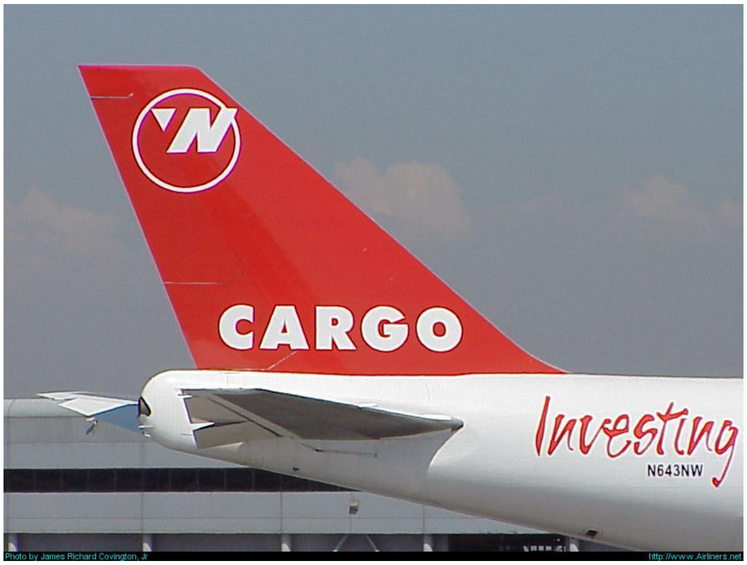


Photo by James Richard Covington, Jr

http://www.Airliners.net

Figure 3: This is a sample of the image that only contains partial aircraft which we will exclude from Maji et al. (2013c)

After we have built the new dataset, it will be divided into training, validation, and test sets. We will split the dataset into 70% training, 15% validation, and 15% test. In order to fully utilize the available data, we will use data augmentation such as rotate, color distort, flip etc, to increase training data and help generalization by learning the internal representation of transformations. We believe that following these steps will create a high-quality and up-to-date dataset that is suitable for training our model.

## 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.

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

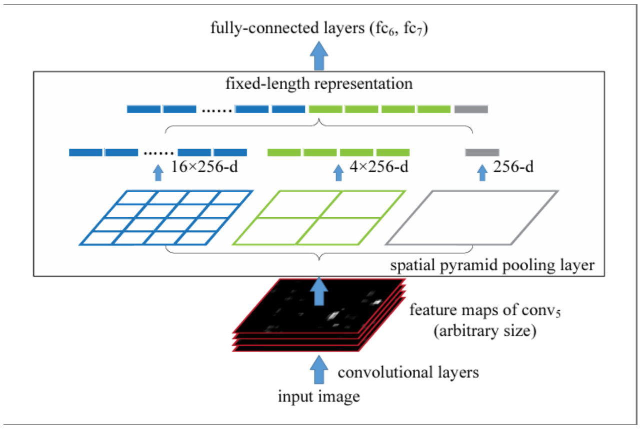

fully-connected layers ($fc_6$, $fc_7$)
fixed-length representation
16×256-d
4×256-d
256-d
spatial pyramid pooling layer
feature maps of $conv_5$
(arbitrary size)
convolutional layers
input image

Figure 4: 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).

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.

In this project, we modify state-of-the-art CNN architectures, namely **VGG-19** Simonyan & Zisserman (2015) and **ResNet-50** He et al. (2015), by inserting an SPP layer between the convolutional and FC layers. **VGG-19** is a deep CNN comprising 19 layers and employs simple, uniform 3x3 convolutional filters to progressively extract features. **ResNet-50** is a 50-layer deep network introducing residual connections (skip connections) to facilitate the training of deeper models by mitigating the vanishing gradient problem. Notably, ResNet-50 uses a *Global Average Pooling* (GAP) layer immediately before the FC layer, which can be considered a special case of SPP with a single 1×1 bin. In our implementation, we replace this GAP layer with an SPP layer, enabling the capture of multi-scale spatial information. We will subsequently compare the performance of the two models and the best-performing one will be chosen as our main model.

# 5 Baseline Model

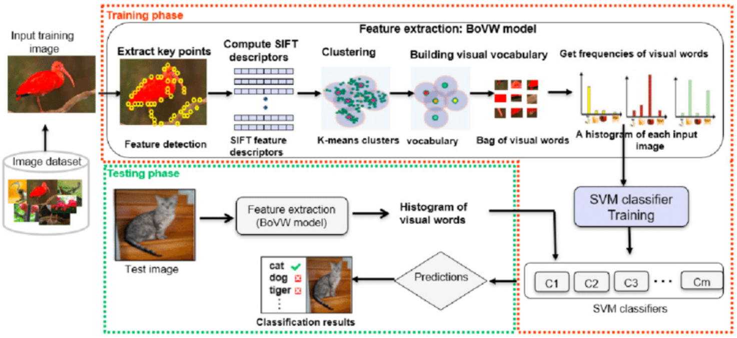


Figure 5: An Illustration of the BoVW+SVC Approach in Image Classification Task. Adapted from Filali & Zghal (2021)

The authors of the dataset used an SVC model as a baseline Maji et al. (2013b). Specifically, they selected a non-linear SVM on a $\chi^2$ kernel, bag-of-visual words (BoVW), 600 k-means words dictionary, multi-scale dense SIFT features, and $1 \times 1$, $2 \times 2$ 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. Since this is the baseline the authors of the dataset used, we would like to continue to include it in this project.

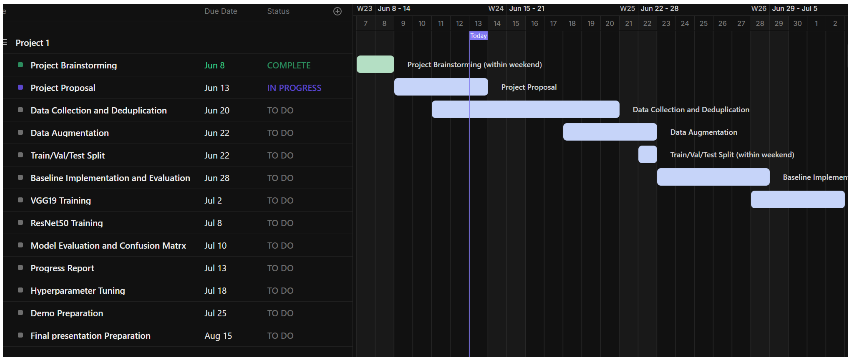
However, since this paper was published in 2013, this traditional approach does not leverage the advancements and power of deep learning networks. To improve performance, we can modify the pipeline by replacing the SVM classifier with a neural network model in the final classification step. We plan to use three FC layers after the BoVW feature extraction to classify the airplane model.

## 6 Project Plan

The team will communicate primarily through a social media group to ensure effective and timely coordination. Weekly meetings will be held to monitor progress, address challenges, and distribute upcoming tasks.

Code development will be conducted on Google Colab for convenience and collaboration, with updates regularly pushed to the team's GitHub repository using version control practices.

Major tasks, assignees, and timelines are organized and visualized in a shared Gantt chart, which is editable by all team members, as shown in Figure 3.



|  | Due Date | Status |
|---|---|---|
| Project 1 |  |  |
| Project Brainstorming | Jun 8 | COMPLETE |
| Project Proposal | Jun 13 | IN PROGRESS |
| Data Collection and Deduplication | Jun 20 | TO DO |
| Data Augmentation | Jun 22 | TO DO |
| Train/Val/Test Split | Jun 22 | TO DO |
| Baseline Implementation and Evaluation | Jun 28 | TO DO |
| VGG19 Training | Jul 2 | TO DO |
| ResNet50 Training | Jul 8 | TO DO |
| Model Evaluation and Confusion Matrx | Jul 10 | TO DO |
| Progress Report | Jul 13 | TO DO |
| Hyperparameter Tuning | Jul 18 | TO DO |
| Demo Preparation | Jul 25 | TO DO |
| Final presentation Preparation | Aug 15 | TO DO |

Figure 6: Gantt chart showing project timeline and major tasks

## 7 Ethical Consideration

This project is intended for educational purpose only. The team will ensure all datasets used are publicly available and no personal or sensitive information is included such as identifiable individuals or private property. The dataset focuses exclusively on publicly available and commercially released aircraft models. No unreleased, proprietary, or classified aircraft data is used. Therefore, the project does not involve any risk of confidential information disclosure.

## 8 Risk Register

Table 1: Project risks, their likelihoods, and corresponding response strategies.

| Risk | Likelihood | Response / Contingency |
|---|---|---|
| A team member withdraws from the course. | 5% | This is considered highly unlikely, as all team members have no conflicting commitments with this course during the summer term. |
| Data collection takes longer than expected. | 50% | A significant part of this project involves manually collecting images from various sources to build a new dataset. We will manage this risk by allocating extra buffer time in our schedule and prioritizing essential data first. |
| Model training exceeds the estimated duration. | 30% | Training the model on over 7,000 images is expected to take approximately 100 hours on 4 A100 GPUs, accessible via Google Colab and cloud services. We also have an offline workstation equipped with an Nvidia 4070 GPU available as a backup. To mitigate this risk, we will closely monitor training progress, utilize multiple platforms in parallel if necessary, and seek access to higher-performance computing resources should delays occur. |
| Busy schedules limit team members' availability. | 5% | This risk will be managed by scheduling regular weekly meetings in advance and assigning clear, manageable individual deadlines. Furthermore, since all team members are part-time students, we generally have sufficient flexibility to accommodate project commitments. |
| The proposed method underperforms. | 40% | The model may struggle to accurately distinguish between aircraft models, as different variants often exhibit only minor discernible features, such as variations in the number of windows. Additionally, factors like image resolution, angle, and lighting conditions could obscure subtle identifying characteristics. To address this, we could either augment the dataset with annotated images focusing on distinguishing features or experiment with alternative model architectures like fine-tuned transformer-based vision models to improve classification performance. |

## 9 Link to Github and Method of Communication

We host our source code publicly on Github (link). We will communicate through weekly online meeting through Zoom and the chat app WeChat.

# REFERENCES

Airliners. Airliners.net — aviation photography, discussion forums news, 1995. URL `https://www.airliners.net/`.

Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 105 (10):1865–1883, 2016.

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`.

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

Zhuang Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning markov random field for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6): 1318–1332, 2018.

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`.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 945–953, 2015.