Title: Identifying aircraft from above

Name: Kai Roper-Blackman

Registration Number: 1602999

Supervisor(s): Adrian Clarke, Sebastian Halder

Second assessor: Luca Citi

Degree Course: BSc Computer Science (G400)

Contents

[Acknowledgements 4](#_Toc5633871)

[Abstract 5](#_Toc5633872)

[List of Symbols 6](#_Toc5633873)

[Project Aims and Objectives 7](#_Toc5633874)

[Primary Objectives 7](#_Toc5633875)

[Additional goals 7](#_Toc5633876)

[Methods 8](#_Toc5633877)

[Internal and External Libraries 8](#_Toc5633878)

[OpenCV 8](#_Toc5633879)

[Scikit-Learn 8](#_Toc5633880)

[Scikit-Image 8](#_Toc5633881)

[FPDF 8](#_Toc5633882)

[NumPy 8](#_Toc5633883)

[Matplotlib 8](#_Toc5633884)

[Tkinter 8](#_Toc5633885)

[Data collection 9](#_Toc5633886)

[Images 9](#_Toc5633887)

[Parsing 9](#_Toc5633888)

[Histogram of oriented gradients 9](#_Toc5633889)

[Feature vector 9](#_Toc5633890)

[Machine learning 10](#_Toc5633891)

[Support vector machines 10](#_Toc5633892)

[Cross validation 10](#_Toc5633893)

[Tuning 10](#_Toc5633894)

[Hyper parameters and decision boundaries 10](#_Toc5633895)

[Data set 10](#_Toc5633896)

[Kernel 10](#_Toc5633897)

[Technical Achievement 11](#_Toc5633898)

[Results 11](#_Toc5633899)

[Project Planning 12](#_Toc5633900)

[Momentum 12](#_Toc5633901)

[Adapting to change 12](#_Toc5633902)

[Identifying and dealing with risks 12](#_Toc5633903)

[Achievement 12](#_Toc5633904)

[Performance 12](#_Toc5633905)

[What have I learnt? 12](#_Toc5633906)

[Conclusions 13](#_Toc5633907)

[Discussion 13](#_Toc5633908)

[References 14](#_Toc5633909)

[Tables, Graphs, Figures and Equations 14](#_Toc5633910)

[Appendices 14](#_Toc5633911)

[Sustainability 14](#_Toc5633912)

[Legal 14](#_Toc5633913)

[Ethical 14](#_Toc5633914)

[Intellectual property 14](#_Toc5633915)

# Acknowledgements

A special thanks to my parents Annie Roper and Seon Blackman for supporting me throughout University.

Thank you to all the academic staff at the University of Essex

# Abstract

We all know what an aircraft looks like, but does a computer? A seemingly simple task that can be carried out by individuals at age two, poses a complex problem to modern technology. Machine learning is a relatively new field with little research but already boasts claim to many applications such as driverless cars and face recognition systems. The development of object recognition is the center of many companies’ business models and objectives, making aircraft identification such an interesting topic to research.

Existing images of ground and aircraft are pre-processed using Histogram of Gradients to create feature descriptors. Feature descriptors describe the orientation of a gradient within an image subsection. Support Vector Machines are passed feature descriptors with labels for training. Once training is completed, the support vector machine accepts a test set and returns predictions. Large image search takes a large image and looks within a smaller area for aircraft. Search area parameters are provided by the user.

The results obtained from cross validation show an accuracy of 100% when identifying standalone aircraft. However, when searching for aircraft in larger images, accuracy drastically decreases to around 55% as some aircraft are overlooked. After optimization, the system used to identify aircraft can be applied to other identification problems with possible military and commercial uses.

# List of Symbols

|  |  |
| --- | --- |
| **Acronym** | **Meaning** |
| SVM | Support Vector Machine |
| PNG | Portable Network Graphics |
| px | Pixels |
| HOG | Histogram of Oriented Gradients |
| PIL | Python Image library |
| GUI | Graphical User Interface |
|  |  |
|  |  |
|  |  |
|  |  |

# Project Aims and Objectives

The original aims of this project have changed since the initial report. This was because of a change of approach to the given problem. The main objective is to allow a computer to differentiate between **aircraft** and **ground** images. This can be broken into several smaller objectives.

## Primary Objectives

1. To gather a set of aircraft and ground images to create a training and test set
2. To normalise the training and test set
3. Change the orientation of aircraft to ensure the face north
4. Resize the images so they can be pre-processed accurately
5. To pre-process the training set and label accordingly.
6. To Train an SVM with the Training set
7. To test the SVM with the test set
8. Further optimise the SVM by tuning decision boundaries
9. Cross-validate the training set
10. Display results of classification to user.

Assuming all primary objectives are completed successfully, additional objectives are to be attempted

## Additional goals

1. To attempt to recognise aircraft in a large image using user defined search criteria and show aircraft locations to user after search is completed.
2. To generate heat maps of large images to identify hot spots
3. To attempt to recognise non-commercial aircraft such as private aircraft and helicopters.

The

# Methods

## Internal and External Libraries

During the project, I have used a variety of libraries to aid the completion of the projects aims and objectives. Writing code to support these functions would be impossible to undertake in the tight timescale given.

The language I have chosen for this project is python as it has a large range of image processing libraries available and offers a wide range of features such as object orientation.

### OpenCV

The library used to read images from PNG format to Numpy arrays was OpenCV. OpenCV has tools for image manipulation and other image related functions. OpenCV was used to rotate images before training.

### Scikit-Learn

Scikit-Learn is a well-documented Machine learning library that offers a wide range of tools for machine learning, data mining and data analysis. Scikit-learn has developed a Support Vector Machine that can be easily utilised for aircraft identification.

### Scikit-Image

Scikit-Image is a collection of image processing algorithms for the Python programming language. This particular library contains the HOG algorithm used in image pre-processing.

### FPDF

FPDF is a library that allows interaction between Python and pdf files. This allows the results of classification to be saved to and viewed by the user.

### NumPy

NumPy is the fundamental package for scientific computing in Python [1]. It comes with a variety of features and functions such as mathematical, logical and shape manipulation. This makes it the perfect library to store images as arrays. Other libraries used during this project utlise NumPy often returning NumPy arrays after specific functions. During the training stages, images are rotated using NumPy’s rotate function.

### Matplotlib

Matplotlib is a 2D plotting library that produces publication quality figures. During this project, I used its features to display results of classification to the user and allow them to interact with them using the libraries GUI. This allows them to zoom in and move Images and plots around as they wished. The user can also choose to save the output if they wish for later viewing.

### Tkinter

Tkinter is the Python standard for GUI development. The library allows the development of complex windowed GUI’s. It allows the placement of buttons, labels, text boxes, images, radio buttons and drop down menus. Tkinter also has GUI’s for functions such as file selection.

### PIL

PIL is python’s image library that adds support for opening and saving images in multiple image formats. The library contains functions to easily manipulate images and draw shapes/ text over them. The shape functions have been utilised in this project.

## Data set

For a computer to recognise an object, first you have to teach it what the object looks like. There are many different learning techniques however, the technique I chose to implement is supervised learning. To carry out supervised learning, a training set must be created. The training set is labelled and provided to the machine learning model.

### Images

The data set currently consists of 250 images of aircraft and 250 images of ground. Images were obtained from google earth by taking screenshots of airports, then cropping aircraft and ground and saving them as individual images. Examples of training images are shown in Figure 1.

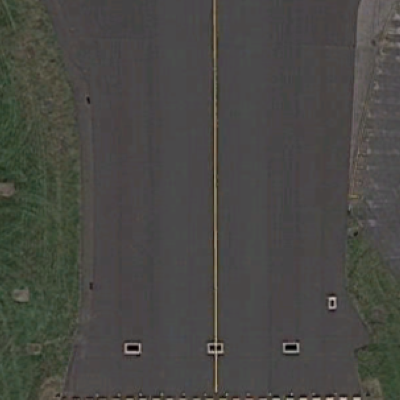


Figure 1: Image of ground (left) image of aircraft (right)

I ensured all images of aircraft were rotated so that the aircraft faced north and were all of the same dimensions (400x400 px) ensuring training was fair and balanced. Images are of the PNG format for simplicity.

### Parsing

An image in computer science is regarded as an array of values. Each field contains a value which is displayed onscreen as colour intensity. To train a machine learning model, images are placed in a large array of arrays. To create this array, images are read from storage using the OpenCV library. The OpenCV function reads the image and returns a NumPy array. The image is then stored in a larger array. This process is repeated adding each image to the larger array.

## Pre-Processing

Why is pre processing important

What happened when images weren’t pre processed.

### Histogram of oriented gradients

### Feature vector

## Machine learning

Training

### Supervised learning

Supervised learning is the process of providing a machine learning model with labelled training data so it gains

### Support vector machines

What is an svm

### Kernel

### Hyper parameters and decision boundaries

### Tuning

Grid search

### Cross validation

What is cross validation and what scores did you get from different data sets?

### Data set

# Technical Achievement

## Results

# Project Planning

## Momentum

## Adapting to change

## Identifying and dealing with risks

## Achievement

## Performance

## What have I learnt?

# Conclusions

## Discussion

# References

[1] <https://docs.scipy.org/doc/numpy-1.13.0/user/whatisnumpy.html>

# Tables, Graphs, Figures and Equations

# Appendices

## Sustainability

## Legal

## Ethical

## Intellectual property