USING MACHINE LEARNING TO IDENTIFY DEPICTIONS OF LIGHTNING IN PAINTINGS – IMPLEMENTATION OF A YOLOV2 OBJECT DETECTOR

Jason D. Ziemons

School of Electrical & Information Engineering, University of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

Abstract: This report entails the implementation of a Convolutional Neural Network (CNN) based system using the Resnet18 architecture for the purpose of detecting depictions of lightning in paintings. A combined model created from an image classifier followed by a You Only Learn Once object (YOLO) detector is used as the final method of image recognition. The image datasets were manually collected, and the object detector model was trained on a collective 3031 natural and lightning painting images. Testing datasets were created as the Louvre dataset which contains 3038 paintings, 14 of which are lightning and the assessment dataset of 500 paintings of which 122 are lightning. As the design focus of this paper the YOLOv2 object detector and final combined model are tested and compared. The object detector achieved an accuracy of 79.4% and 51.7% for the assessment and Louvre datasets respectively, while the combined network had an accuracy of 81.2% for the assessment data and 96.6% for the Louvre images. Results regarding the Louvre dataset are used in determining Benjamin Franklins influence on French culture; however, no significant outcome was observed from the model.

1. INTRODUCTION

Image recognition is often based on a form of machine learning (ML) known as deep learning which can automate or simplify complex tasks. This report will present the implementation of a CNN for the task of image processing in a painting medium. The primary goal will be to identify whether a painting has any depictions of lightning and consequently sort these images apart from any others. Through application of the network a link between Benjamin Franklin and his fame in lightning science will be used to identify if his presence in France was able to impact the art culture. An investigation into the dataset creation, image pre-processing, object detection and resultant outputs will be explored in order to test the model's viability. The final implementation is compared using a historically sourced dataset of paintings from the Louvre Museum and a collection of lightning paintings gathered from various museums.

2. BACKGROUND

Machine learning algorithms have become an increasingly common method of image classification in visual processing systems in the modern world. They hold a significant utilization in identifying objects where image clarity is conventionally dictated by camera resolution only. By exploring ML as a method of painting analysis it represents a domain shift problem whereby additional challenges are encountered. In painting analysis not only is image quality a factor but also the obscurity of the painting, historical representation of an object and the artists style of painting [1].

By placing emphasis on specific painting focal points using ML models art historians and curators would be able to make classification in paintings simpler. In the use of art curators an advanced ML system would allow paintings to be catalogued automatically. This will therefore refine search options to allow paintings to be more easily found based on object representations [1]. As that general ML model would dictate a significant undertaking a version based on lightning identification is looked at in this paper. To indicate the use to art historians the hypothesis of Benjamin Franklin's influence in France acts as a functional example.

2.1. Historical considerations

As the research topic under investigation is of a historical context it is important to consider the socio-political atmosphere of France in the time period. Benjamin Franklin lived in France between 1776-1785 which was a very crucial period due to the onset of the French Revolution in 1789. The importance of this event with respect to art history is that iconoclasm spread throughout France and paintings were targeted due to their symbolism of wealth [2]. These events do present a unique condition with respect to gathering a test dataset as in an attempt to protect artworks, many paintings were stored in the Louvre Palace [3]. Images gathered from the Louvre Museum are therefore the main consideration for a testing dataset as they are time period specific [4].

2.2. Technical considerations

The primary technical considerations that are defined for this ML implementation is the use of MATLAB (R2021a) as the programming language. In order to execute the CNN architecture and environment using a transfer learning approach the Deep Learning and Parallel Computing Toolboxes had to be installed in order to access MATLAB's ML functionality. The Parallel Computing Toolbox provides access to GPU support allowing for faster training of the model. A Windows based laptop with a NVIDIA GeForce MX250 GPU was used as well as the Linux based Jaguar compute cluster which has a NVIDIA Tesla K40c GPU.

2.3. Literature review

Image recognition systems are based on a combination of the technical code implementation as well as extensive understanding of the dataset domains and pre-processing used.

Crowley [1] investigated a similar research topic over a broader scope of artwork and included multiple objects as a focus of detection. A comparison between shallow and deep networks was made and it was determined that the deep networks were able to outperform shallow models. It was also determined that the networks trained solely on images of paintings were able to make classifications more accurately than those trained on natural images. Based on the CNN networks tested the Resnet 152 architecture was able to achieve the highest accuracy with paintings [1].

Stromp, *et al* [5] presented findings based on the representation of lightning in paintings throughout history. It was determined that humans are limited in remembering fine details about lightning as it is visible for a brief moment. This would indicate that lightning representations before photographic images have fewer details and therefore differ in shape compared to modern depictions [5]. As these painting interpretations of lightning are limited it may introduce underfitting into the model resulting in a lower accuracy.

Convolutional Neural Networks have different forms of implementation based on how the output image is determined. Saponara, *et al* [6] made use of a You Only Learn Once (YOLOv2) object detector with a MATLAB implementation that identifies the image and localizes the object. It was determined that the YOLOv2 object detector was able to perform significantly faster than a R-CNN detector and achieve a high accuracy. The steps taken for image pre-processing are also indicated using MATLAB's ground truth labeller app for annotating the images before training [6].

Cui, *et al* [7] demonstrates the benefits that can be obtained in using the transfer learning approach for training a CNN on images in a similar domain. Emphasis is placed on the network having a beneficial output when using smaller datasets with the transfer learning approach compared to an untrained network. Additionally, a relationship is created between the quality of image resolution and obtaining a higher overall accuracy [7].

3. DATA COLLECTION AND AUGMENTATION

The data collection aspect of any machine learning algorithm presents as one of the largest factors to a project's success. As a CNN effectively utilizes a black box model when determining features of an object's depiction, the quality of input data will considerably impact output performance [8]. Images for general objects are frequently organized into datasets, yet lightning images being a more specific use case did not appear to have any available datasets. All images used in the training and testing of this ML model were therefore manually collected through different websites.

3.1. Dataset collection

The images collected for use in the datasets comprise of 3 different categories and encompass 2 image domains. Training data was collected as a combination of natural images and paintings for use in the model. This is because there were very few paintings of lightning easily available in the public domain. These images were obtained primarily through google images, a small dataset found on Kaggle, and other open-source websites such as Unsplash, Pixabay , Pexels and FreeImages . As it was highly likely that duplicate images had occurred a software called VisiPics was used in order to filter out and delete these images [9]. A total of 3031 unedited images was gathered for the training dataset. As an image classifier requires at least 2 types of images for detection as a part of training data another dataset of random images was created. These images are from the ILSVRC2011 validation set found in the ImageNet database, it contained a total of 50000 images of which 10-20 thousand were used in testing [10].

Testing datasets encompass the Louvre Museum images mentioned earlier which consist of 3038 images and which will be referred to as the Louvre dataset. This dataset was manually searched for images of lightning and only 14 images are present. As this dataset has a deficit of lightning paintings an additional dataset was created of fine art paintings gathered from different museums. A total of 500 artworks was included in the dataset with 122 of those having varying depictions of lightning. This will be referred to as the assessment dataset.

3.2. Image pre-processing

By editing or creating variations of the images it allows different features to be more easily observed by the ML model or can remove unnecessary noise from an image. Methods used to try and improve performance were applying filters to training data and correcting perspective distortion in test data. Additionally, steps were taken to examine the effect of converting images into their greyscale and black and white variants. This proved to be ineffective however as by using a transfer learning approach it meant that considerable modifications needed to be made to a model pre-trained on coloured images.

As a way of increasing the number of test data images that could be used in training an oil painting filter was applied that doubled the testing data to 6062 images, which was used for the image classifier. An additional benefit to the filter is that the training data now had images that could more closely be compared to the painting medium. In the testing data an issue was noted whereby the photographs of the paintings were taken from different angles. As the training data images were forward facing this would introduce noise and result in incorrect image detections [11]. The images were manually altered using a MATLAB downloadable code called Perspective Control/ Correction [12]. Figure 1 below shows the changes made to distorted images using this method.

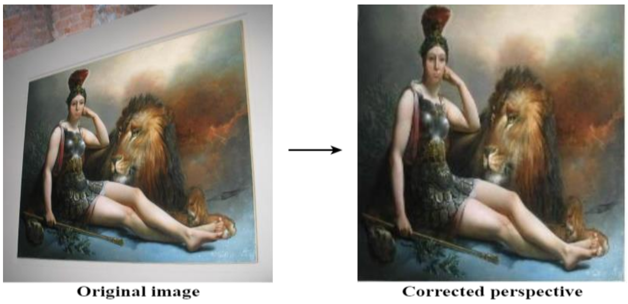


Figure 1: Demonstration of applying perspective correction methods to a painting [13].

Lastly, in order to use the images with an object detector they have to be annotated with a label and a bounding box is created over the regions of lightning in the image. This process was done manually using MATLABs Image Labeller app.

3.3. Data augmentation

Data augmentation is a process completed in the main source code of the object detector that creates different variations of the same image. This is done by applying rotations, reflections and contrast changes that scale up the number of images used for training. A further benefit to this process is that the network is exposed a broader range of possible image features allowing a higher accuracy [14]. Additionally, the images have to be resized to a resolution of 224 x 224 to be passed into the network and any grayscale images converted to colour.

4. PROPOSED SOLUTION

The final image recognition system is based on a combination of an image classifier using a Resnet 18 CNN and YOLOv2 object detector with a Resnet 18 architecture. By combining the two networks the image classifier acts as a filter which can remove a large number of false positive images. These remaining images are then passed into the object detection system which removes additional images and label's locations that it identifies as lightning. Other than fewer false positives this method also allows the images to be examined at a lower identification threshold for determining more abstract lightning depictions. This section is primarily focused on the object detection system and not the image classifier.

MATLAB has functionality for three types of object detectors, the RCNN (regions with convolutional neural networks) variety, YOLO and SSD (single shot detection). From these three detectors the YOLO version was determined to have a significantly faster training and processing time [6]. It was also found in a research paper that YOLO was able to achieve a higher accuracy when transitioning to artwork compared to the other detectors[15]. The YOLOv2 detector is a later version that is able to achieve a higher accuracy and image recall.

The YOLOv2 detector works by replacing the final few layers of the Resnet18 architecture with its own layers. It divides an image into a grid of smaller regions that are examined for objects and a bounding box for each region is selected. The box that is given the highest score or confidence is determined as the closest representation of lightning in that image [6]. Figure 2 below presents the working principle of the entire object detection system from dataset modification, through the most common Resnet18 layers and ending with the YOLOv2 detector output. The MATLAB code that is implemented was adapted to fit the lightning parameters based on the original source code from a MATLAB example solution [16].

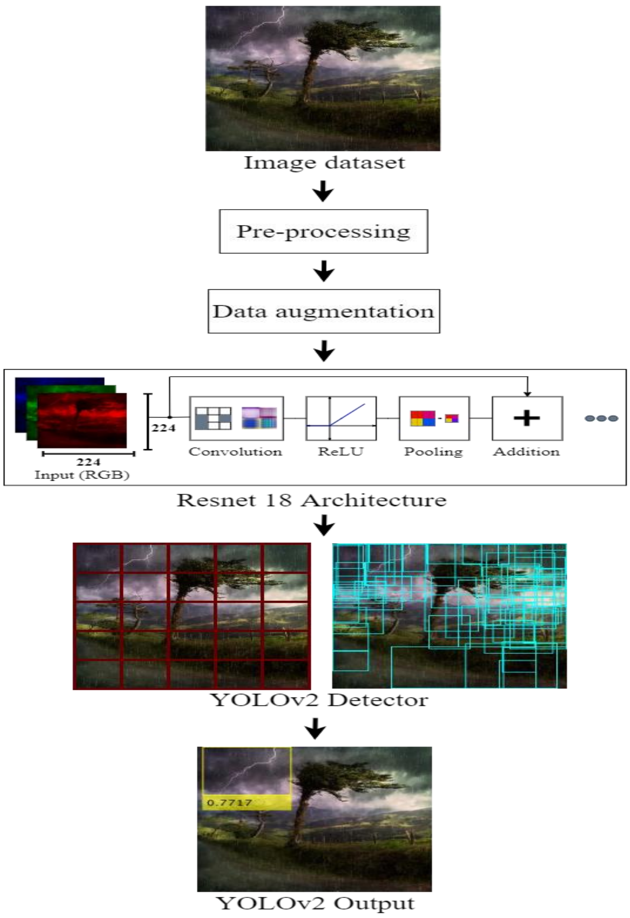


Figure 2: Complete process of the final implementations object detection system.

After determining the object detector's structure, the main course of action is modifying the networks hyper-parameters to improve output accuracy. As every image is different there is no clearly defined set of hyper-parameters that will improve performance for each model and must be determined through a trial-and-error approach. In testing parameters for the object detector networks were run with variations in the training parameters and the output data was correlated based on the accuracy found in the test datasets. Table 1 below represents the changes tested for training parameters and which parameters had the best identified output. Table 1 in Appendix D has specific details regarding each tested network.

Table 1: Choice of training parameters for YOLOv2 object detector.

Parameter	Values tested	Highest accuracy
CNN network	- Resnet 18 - Darknet-19	Resnet 18
Mini-batch size	- 16 - 32 - 64	32
Number of epochs	- 5 - 10 - 20	20
Learning rate	- 0.001 - 0.0001 - Piecewise decay of 0.5 for above 2 values	0.001
Optimizer	- Adam - Sgdm - Rmsprop	Adam
Number of anchor boxes	- 1 - 10	10
Feature layer	- res2b_relu - res3b_relu - res4b_relu - res5b_relu	res5b_relu

5. RESULTS

When determining the output of an image recognition system the results are often represented as a metric of precision and recall. The precision is a measure of how accurately the network is able to detect lightning, what proportion of images identified are actually correct. The recall determines how well the positive images were actually found, were all images with lightning identified. Based on the output images four variables can be classified, the True Positives (TP), False Positives (FP), False Negatives (FN) and True Negatives (TN). True values were accurately classified according to their class and false values were incorrect predictions [17].

Equations 1,2 and 3 below are how precision, recall and overall accuracy are determined.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{3}$$

Using the testing datasets, a confusion matrix for the object detector and final combined system was created, shown in Figure 3 below. This diagram displays the number of images that were correctly and incorrectly identified. For the Assessment dataset a confidence level of 0.5 was used for both models and for the Louvre images a confidence of 0.3 is used. This is due to the images in the Louvre dataset being weaker representations of lightning and therefore more difficult to detect.

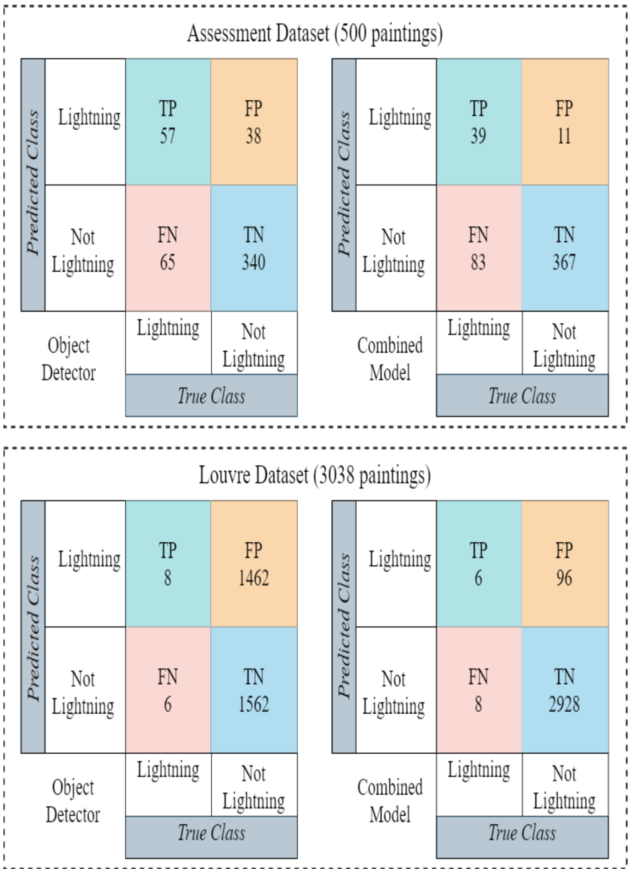


Figure 3: Confusion matrices of Object detector (left) and combined network (right) for assessment and Louvre dataset.

By applying the equations above to the data presented in figure 3, the 2 networks accuracies can be determined. These values are shown in Table 2 below.

Table 2: Results obtained from the testing datasets for the YOLOv2 object detector and combined model.

Object Detector		
Measurement	Assessment dataset	Louvre Dataset
Precision	60.0%	0.5%
Recall	46.7%	57.1%
Accuracy	79.4%	51.7%
Combined Model		
Measurement	Assessment dataset	Louvre Dataset
Precision	78.0%	5.9%
Recall	32.0%	42.9%
Accuracy	81.2%	96.6%

From the results above it is evident that the object detector is able to achieve a higher recall and identify more lightning depictions. It also manages to classify more non lightning images as lightning, meaning it has a lower accuracy and precision value. The combined network solves this issue by creating a balance between the recall and precision which results in a higher overall accuracy [17].

From the Louvre dataset using the combined network it is now possible to segment the images based on time periods in line with Benjamin Franklins time in France. These results are indicated in Table 3 below and show that there are more depictions of lightning in the 100 years after Benjamin Franklin's time in France.

Table 3 : Lightning images identified in Louvre dataset separated by time period.

Time period	Number of images	Lightning images	Total images found
1676 - 1775	974	2	33
1776 - 1785	200	0	10
1786 - 1886	1864	4	59

Although the results in Table 3 above suggest an increase in lightning images it should also be noted that there are more images in the 1786 – 1886 period in general. This means that the data is skewed and from a probabilistic viewpoint there would be an expected higher number of lightning images in this time period [18].

6. CRITICAL ANALYSIS

Based on the final results obtained for the YOLOv2 object detector and combined network. A critical analysis of the final implementation and design decisions implemented will be conducted in order to determine the model's overall success.

6.1. Evaluation of design

The implemented design is able to perform well under the condition that the lightning depiction closely resembles realistic lightning. When introduced to depictions of lightning that are not the focal point of the painting such as those found in the Louvre dataset, a lower confidence has to be used. This indicates that the training data did not consist of enough images that depicted background lightning and the model is not able to adjust well to these types of paintings.

By implementing the combined image classifier and object detector network a valuable design decision was made that resulted in fewer not lighting images being identified as lightning. Overall, the model is able accurately predict approximately 40% of the lightning images in both datasets with a limited number of not lightning images. This implies that the user will still have to manually search through the final set of images, but it is considerably less images than the original dataset.

6.2. Areas of concern

Due to the use of natural images in the training dataset being more prominent, the model is more suited to determining lightning in photographs over paintings. Resulting in fewer lightning paintings being detected overall. The skewed data in the Louvre dataset means that substantial results can not be obtained on Benjamin Franklin's influence in France with such few depictions of lightning. In the data augmentation stage of the images, they have to be resized to a lower resolution to be used in the CNN. This resulted in detail being lost in the depictions of lightning and no methods were used to correct these issues which was a probable cause of smaller depictions being harder to detect.

6.3. Future improvements

To improve upon the current model, steps would need to be taken to increase the number of training images for a broader range of features learned. This could be done by extracting images from video sources as well as contacting lightning photographers for access to their images. To balance the Louvre dataset, it would be useful to explore museums with more historical paintings originating from France. By making modifications to the Resnet 18 architecture it would be possible to change the image resolution requirements and obtain a higher accuracy.

7. CONCLUSION

A solution to detecting depictions of lightning in paintings has been presented using the combination of an image classifier and YOLOv2 object detector. The network is able to accurately detect clear representations of lightning but has a decrease in performance as the lightning looks less like a natural representation. Using only the object detector an accuracy of 79.4% was obtained for the assessment dataset and 51.7% for the Louvre dataset. When using the combined system an accuracy of 81.2% for the assessment and 96.6% for the Louvre datasets was observed. When examining the Louvre images with respect to the Benjamin Franklin hypothesis, there is an increase in lightning paintings in the period after he left France. However only 6 images were determined by the network and the data is skewed meaning the results are inconclusive.

ACKNOWLEDGEMENTS

The author would like to acknowledge Sansha Gupta in their contributions to the design , implementation, documentation and presentation of this project.

REFERENCES

[1] E. Crowley, "Visual Recognition in Art using Machine Learning", Doctorate, University of Oxford, 2016.

[2] D. Gamboni, *The destruction of art*, 1st ed. London: Reaktion Books Ltd, 1997, pp. 9-10.

[3] A. Sido, "Making History: How Art Museums in the French Revolution Crafted a National Identity, 1789-1799", Undergraduate, Scripps College, 2015.

[4] "Louvre Museum Official Website", *Le Louvre*. [Online]. Available: https://www.louvre.fr/en. [Accessed: 15- Oct- 2021].

[5] G. Horvath, "How realistic are painted lightnings? Quantitative comparison of the morphology of painted and real lightnings: a psychophysical approach," *Proceedings of the Royal Society A*, vol. 474, no. 2214, 2018.

[6] S. Saponara, A. Elhanashi and A. Gagliardi, "Real-time video fire/smoke detection based on CNN in antifire surveillance systems", *Journal of Real-Time Image Processing*, vol. 18, no. 3, pp. 889-900, 2020. Available: https://doi.org/10.1007/s11554-020-01044-0. [Accessed 15 October 2021].

[7] Y. Cui, *Large Scale Fine-Grained Categorization and Domain-Specific Transfer Learning*. pp. 4109-4116.

[8] J. Brownlee, *Data Cleaning, Feature Selection, and Data Transforms in Python*, 1st ed. 2020, pp. 3-8.

[9] *VisiPics*. 2013.

[10] Olga Russakovsky, **ImageNet Large Scale Visual Recognition Challenge**. *arXiv:1409.0575*, 2014.

[11] S. Bavikadi and V. Bharath Botta, "Estimation and Correction of the Distortion in Forensic Image due to Rotation of the Photo Camera", Masters, Blekinge Institute of Technology, 2018.

[12] M. Chan, *Perspective Control/ Correction*. MATLAB, 2012.

[13] F. Joseph, *La Force ou la Valeur*. Belfort: Louvre Museum, 1817.

[14] B. Zoph, E. D. Cubuk, G. Ghiasi, T. Lin and J. Shlens, "Learning Data Augmentation Strategies for Object Detection", *Arxiv*, vol. 1, pp. 2-4, 2019. Available: https://arxiv.org/pdf/1906.11172.pdf. [Accessed 25 October 2021].

[15] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, *You Only Look Once: Unified, Real-Time Object Detection*. 2021, pp. 779-786.

[16] "Object Detection Using YOLO v2 Deep Learning", *MathWorks*. [Online]. Available: https://www.mathworks.com/help/vision/ug/train-an-object-detector-using-you-only-look-once.html#ObjectDetectionUsingYOLOV2DeepLearningExample-10. [Accessed: 07- Oct- 2021].

[17] J. Hui, "mAP (mean Average Precision) for Object Detection", *Medium*, 2018. [Online]. Available: https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173. [Accessed: 25- Oct- 2021].

[18] R. Sharma, "Skewed Data: A problem to your statistical model", *Medium*, 2019. [Online]. Available: https://towardsdatascience.com/skewed-data-a-problem-to-your-statistical-model-9a6b5bb74e37. [Accessed: 27- Oct- 2021].

APPENDIX A – Personal reflection

Through the course of this project the group dynamic between me and Sansha Gupta was good, and we always managed to keep an open line of communication, by keeping each other updated on our current tasks. Due to the challenges of the Covid-19 pandemic and my car having broken down recently it meant that we were not able to meet as often as we would have liked. We solved this issue by having virtual phone calls over Discord and keeping in contact over WhatsApp. Every time after our weekly meetings with our supervisor we would call each other on the Discord platform and discuss our plans for the upcoming week and assign roles for any major tasks. Whenever one of us completed a task to do with any code, we would upload the data to GitHub so that we always had a backup of our information and could see what was happening from their side.

I feel as though we worked well together and would take the time to try and rationalize any problems that were experienced in the work or explain anything that the other person did not understand. In the beginning we learnt about machine learning together by taking the courses offered on MATLAB's training websites. The course put us in a good place of understanding as we were on equal footing in an unfamiliar subject. We could therefore work together in making our first basic image classifier. We appeared to have a common goal and wanted to meet a strict set of success criteria each week to make sure that we were making progress. The only direct complications we did seem to have when discussing work was occasionally reaching a brief impasse. If one of us suggested a possible path to take and the other disagreed, neither of us was willing give up our argument. These instances were quickly resolved however as we would more thoroughly discuss a point of concern and determine the best decision.

When we were granted access to a GPU through the Jaguar cluster by Professor Ken Nixon which was a Linux system. I was very unfamiliar with using a command line, but she had experience from a super computing competition. We therefore decided that I would take care of the image datasets for the time while she worked on figuring out how MATLAB works with Linux. I appreciated in this moment that we were able to separate the task so easily and agree on a course of action that would quickly advance our progress. We both seemed to be confident in one another's abilities and didn't feel the need to constantly check in to make sure the other was working. Although we were still able to communicate well, the linear nature of the project made it difficult to work on any task simultaneously. We therefore relied on the process of subdividing tasks based on who we felt would be able to handle it better or who had more experience in the subject at that point. From this point forward I was responsible for the datasets and uploading them to a google drive for both of us to use. She was entirely responsible for working with the cluster and running any larger networks we wanted to test.

As for later progress into the project we kept using the system of assigning tasks and we reached the next big implementation step. We both felt uncertain if the image classifier was actually detecting lightning or just a sky and cloud scene. So, we went to work on different methods, she started to work on a random forest algorithm, and I began implementing an object detector. I feel as though our styles of working seem to align in that we both like to try and reach a common goal, yet it doesn't necessarily have to be together. By being able to work on two methods at the same time and being confident that your partner would get it done we could make good progress each week. As long as we constantly kept each other updated and could determine our next steps we were in a good place. Overall, I don't feel as though we had any issues in working on this project together and were able to come together effectively as a team. The amount of work we handed out each week seemed to be fairly balanced and neither of us seemed to be overburdened.

Generally, I am somebody that really dislikes group work because there is usually an imbalance in the distribution of work but having a partner who I knew would be able to handle any task was refreshing. We were also like minded in the way that we went about solving problems and presenting our work so that we would both be happy with the outcome. It was a great experience being able to learn the subject of machine learning without feeling any hinderance from my partner as I could really enjoy what I was doing. We pushed each other to get tasks done at every point in this 6-week period and ultimately ended up with decent outcome that I'm happy we were able to achieve.

APPENDIX B – Project specification

School of Electrical and Information Engineering
University of the Witwatersrand, Johannesburg
ELEN4002/4012: Project Specification Outline

Project No. & Title:	21B31 – Using machine learning to identify depictions of lightning in paintings		
Group Number:	Group 54	Supervisor Name:	Estelle Trengove
Student Name A:	Sansha Gupta	Student Name B:	Jason Ziemons
Student Number A:	1619757	Student number B:	1724762

Ethics:

✓ Request for waiver (does not involve human participants or sensitive data)
☐ Copy of ethics application attached (Non-medical) – School Committee
☐ Copy of ethics application attached (Medical) – University Committee

Supervisor Signature

Project Outline: *(give a brief outline, including the investigation methodology, such that ethics reviewers understand what will be done, and whether or not human participants will be involved, 100 words maximum)*

The aim of this project is to evaluate the impact of Benjamin Franklin on French culture, regarding his research in lightning science. This will be done by identifying depictions of lightning in paintings between 1776 and 1785, using a subset of computer vision called image classification. Image classification refers to the task of identifying and labelling objects in an image using machine learning (ML) algorithms. Thus, an appropriate algorithm must be identified. Additionally, a suitable dataset of fine art must also be found, which can be used to train and test the ML model. No human participation is required.

Project Specification:

The primary objective of the project is to select and train a machine learning algorithm that can identify representations of lightning within a painting medium. This serves to determine the effect that Benjamin Franklin's fame in lightning science may have had on French culture. This aim's deduction is dependent not only on a direct analysis of the algorithm's statistical findings but also on considerations in sociopolitical factors such as the French revolution which resulted in a significant destruction of art. The main considerations to achieve this are discussed below.

Dataset: As the projects primary hypothesis statement is rooted in a historical context it will be important to view the main testing dataset from the same perspective. Shortly after Benjamin Franklin left France in 1785, the French revolution began which subsequently resulted in the conversion of the Louvre palace into a museum of art. This presents an opportunity to use the multitude of paintings housed in the Louvre that also belong to the relevant time period. Training datasets can be obtained by crawling Google Images and Bing Images using python software libraries such as Selenium and Bing image downloader, respectively. Additionally, research papers have been found that cite the paintings and image datasets that were used in that research which can be used as test datasets [1].

Lightning classification: To detect lightning, the dataset will need to be pre-processed to highlight certain characteristics of the lightning, which will help in training the ML model. [1] have identified three features, namely the length of the main branch, the number of additional branches and the "zigzagness" of the main branch.

Algorithm and architecture: [2] provide a thorough comparison between various ML algorithms for image classification in art. Due to the lack of annotated datasets in art, the chosen ML algorithm must be trained on a combination of photographs and paintings containing lighting thus, it must be able to perform across both domains. Based on the datasets available, unsupervised classification algorithms and Convolutional Neural Networks (CNNs) are the most viable option. Additionally, another factor in choosing the algorithm is the size of the dataset, since CNNs require large datasets for training. The following are suitable architectures: VGG-16, AlexNet, ResNet and AlgoNet.

Software considerations: MATLAB and Python are the two main programming languages which are most frequently used in image classification. Each has well documented libraries that can be used in implementing different machine learning algorithms. MATLAB has access to resources like the Deep learning toolbox and interoperability between other open-source programs. Python most commonly uses TensorFlow, Keras and OpenCV.

Milestones:

Project Planning Stage

1. **Obtain datasets**: A sufficient set of images has been gathered for each phase of implementation. Their quality, size, context and quantity have all been prepared to match the objective.
2. **Language and algorithm selection**: Choice of the programming language, algorithm and the supporting resources that will be used in implementation have been determined.
3. **Obtain functional understanding of algorithm:** Analysis into the implementation of the algorithm has been completed and a general structure has been defined.
4. **Characterize identifiers of lightning:** Unique classifiers for lightning identification have been established in terms of the algorithms expected structure.

Project Implementation Stage

1. **Implementation:** A program that can accept images from the dataset and apply the selected constraints has been developed. Training datasets are applied to the software.
2. **Evaluate testing results:** Testing dataset has been applied to the algorithm and results of a reasonable accuracy have been achieved.
3. **Parameter modification testing:** Effects of altering programs lightning characterizing filters have been observed and most effective testing results are utilized for design.
4. **Report on findings:** The historical testing dataset is applied to the algorithm and conclusions on the influence of Benjamin Franklin in France have been established. All processes and results can now be documented in the final report.

Preliminary Budget & Resources:

Available resources

Hardware

- Laptops that have CPUs which would be able to run a machine learning algorithm.

Software

- MATLAB license
- Open-source Python software
- Machine learning libraries:
  - Deep learning toolbox (MATLAB)
  - Parallel computing toolbox (MATLAB)
  - TensorFlow (Python/MATLAB)
  - Keras (Python)
  - OpenCV (Python)

Requesting resources

Hardware

- Access to a Wits computer with a GPU, as it can significantly increase processing times.
- Sufficient storage in computer for storing dataset (preferably SSD)

Software

- None

Budget

- None

Risks / Mitigation:

Risk	Response	Action
Training the data with paintings and pictures can lead to inaccuracies in the algorithms final results.	Avoidance	Using an algorithm which is domain invariant therefore can accurately classify both paintings and pictures to the same degree.
Undertraining the ML algorithm will provide results with a greater error in lightning detection.	Prevention	Have an excess of training data.
Overtraining the ML algorithm can result in the program learning the statistical noise.	Mitigation of impact	Using a method called early stopping, when the results of the algorithm begin to worsen it should not be trained further.
Further lockdown restrictions preventing campus access.	Mitigation of impact	Continue working on the project remotely as well as have personal backup copies of the datasets and code.

1. REFERENCES

[1] G. Horvath, "How realistic are painted lightnings? Quantitative comparison of the morphology of painted and real lightnings: a psychophysical approach," *Proceedings of the Royal Society A,* vol. 474, no. 2214, 2018.

[2] E. J. Crowley, "Visual Recognition in Art using Machine Learning," 2016.

APPENDIX C – Project plan

USING MACHINE LEARNING TO IDENTIFY DEPICTIONS OF LIGHTNING IN PAINTINGS

School of Electrical & Information Engineering, University of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

Jason Ziemons (1724762) and Sansha Gupta (1619757)
Group 54

2 August 2021

ABSTRACT

The project entails identifying depictions of lightning in paintings during 18^{th} century France to understanding the cultural impact of Benjamin Franklin on France during that time. An image classification model based on Convolutional Neural Networks is proposed for this. The ResNet18 architecture is chosen due to its structure which relies on a *residual learning* framework which minimizes degradation in the testing error, thus improving its accuracy. The dataset required to learn the model consists of the training dataset, the validation dataset and testing dataset. The datasets will be procured using various website crawling methods on Google Images and the Louvre website. A project schedule is

outlined spanning the 6-week period over which the project will take place starting on the 13 September 2021 and is divided into four stages of progression for its successful completion.

1. INTRODUCTION

Benjamin Franklin is a distinguished American academic that lived in Paris during the 18th century. The purpose of the project is to quantify his impact on French culture by identifying lightning depictions in paintings from that time period using machine learning (ML).

The aim of this report is to present the methodology which will be used in implementing an image classification ML model for this task. The report consists of a literature review discussing various methods for image classification in paintings, the various datasets required to teach the ML model, an overview of the architecture of a Convolutional Neural Network (CNN), the procedure used in testing the ML model as well as a summary of the project management tasks such as scheduling, cost, resources and risks.

2. BACKGROUND

Machine learning (ML) has become an ever-growing tool in the technical field by applying complex identification models for an efficient analysis of data. It allows the user to create code which is able accurately determine the specific details of a set of data or tangible object with limited feedback required from the programmer. This makes ML incredibly useful for image identification in order to determine instances of a particular object or environment.

In the confines of the art world the most common use cases of machine learning integration appear to be art generation and painting or artist identification. In this project's investigation the main focus will be based on specific feature identification which is intended to identify depictions of lightning found in paintings. To achieve this machine learning methods will be employed in order to allow a large input of painting data to be analysed and any lightning depictions to be identified. Through these identified paintings a conclusion can then be drawn on the impact that Benjamin Franklin's presence in France may have had on French culture due to his fame in lightning science.

2.1. Initial project constraints

In the process of selecting the plan for the intended solution it is important to note that no constraints have been applied to the selection of programming languages or machine learning methodology. The only pre-defined conditions that have been stipulated are that machine learning must be used and an image dataset will not be provided. The investigation of this subject does not involve human participants and therefore does not require ethical clearance.

2.2. Specifications and technical considerations

MATLAB will be used as the programming environment as it has an extensive library of machine learning functions known as the Deep Learning Toolbox. This offers both a code-based implementation of the ML model as well as graphical model creator called the Deep Network Designer. Additionally, processing times may be improved with the use of the Parallel Computing Toolbox if a GPU is available [1].

In selection of a machine learning algorithm many considerations have to be made to ensure the best possible results are achieved. One of the most prominent aspects relates to the number of images that will be gathered as certain machine learning architectures can be highly susceptible to a data deficit. So, if it is possible to gather thousands of images then a convolutional neural network (CNN) presents as one of the better choices, as the features of lightning will be determined through training. If only a few hundred images are obtained, then a more general classification method can be used but will result in the programmer needing to identify lightnings characteristics which can sway results to their viewpoint [2]. These two descriptions present the classification between deep or shallow machine learning methods and requires insight into the expected yield of images.

Further considerations relating to the achievability of a larger dataset is deciding whether to only include paintings of lightning when training or both paintings and photographs. This will both improve results as more images can be obtained but may also have a skewing effect on what the programme is looking for. This is due to a domain shift between the visual representation of lightning from an artistic viewpoint and the overall realism of the photograph. It will also be influenced by the texture and quality obtained in a painting compared to a photograph which may present very different backgrounds and identifiers [3].

3. EXISTING WORK

Various studies have been done to perform image classification in paintings using various machine learning models however, very few focus on task of classifying paintings by object classes [3]. Rather, studies have focused more on the classification of paintings by genre, artist or style [4, 5]. In most instances a combination of natural images (i.e. photographs) and paintings is used to learn the program showing successful results.

Crowley [3] undertakes the task of classifying different artworks based on the PASCAL VOC dataset which contains 20 classes such as aeroplane, bird boat, etc. It compares both shallow and deep representations of images and it is found that deep representations perform better overall at the task of image classification. It also demonstrates that the ResNet152 architecture outperforms other deep architectures like VD-16 and VGG-M at the task of minimizing the domain shift problem.

Zhao et al. [4] classifies paintings by style, genre and artist using ResNet50 and other ResNet variants such as Res2Net, ResNeXt, etc. In total, 7 different models are compared using transfer learning on 3 different art datasets. Transfer learning refers to the process of using pretrained models and fine-tuning its parameters to suit the domain of choice. The study concludes that the models pretrained using ImageNet are the most successful in art classification resulting in an increase in accuracy of ≈9% when compared to models trained on randomly assigned weights. Additionally, using pre-trained parameters was found to be more impactful when using smaller datasets vs. larger datasets.

4. TRAINING DATASET

As discussed previously, the dataset plays a vital role in the creation of a successful machine learning algorithm and must therefore be planned for accordingly. The dataset will be divided into three categories based on phases that the machine learning programme will be in. The largest division of images will be allocated to the training of the network in order to establish what the programme will be looking for. These will constitute entirely of lightning images, so the network knows to look for similar depictions. Next, a validation dataset will be used to test that the programme is actually able to identify images that display lightning. This dataset will consist of various images include definite depictions of lightning to test that these can be differentiated from the other images. Finally, a historical dataset of paintings from Benjamin Franklin's time period of French origins will be tested in order to determine if he had an effect on Frances's art culture.

4.1. Training and validation datasets

The training and validation images can be any pictures of lightning that are obtained from the internet as long as they consist of a large number of paintings. To determine these images the proposed method is to use a python script which allows an instance of google chrome to be opened using a web driver and the images automatically downloaded. There are many sources online that present this method using a python library known as selenium which can take a screenshot of the desired image and save it directly to a pre-defined file on the computer [6]. The code simulates a user by opening chrome, searching the image prompt and scrolling down all available images until they are exhausted. This method allows all or a specified number of images to be saved quickly without manually needing to save every image individually. It is mentioned that not all images will be perfect, meaning there may be some loss to this method.

4.2. Testing dataset

The primary test dataset that has been identified is based on the painting collections presented in the Louvre Museum. These images are ideal as the collections recently became open to the public, they are historically accurate and present a few thousand images for testing. Additionally, the Louvre Museums website has image filters in place allowing the selection of only paintings, specific time periods and the selection of French artists. By applying these filters around the 1776 and 1785 time periods, the datasets before, after and during Benjamin Franklin's time in France can be obtained.

To obtain these images a simple Google Chrome extension called "Download all images" can be used. It will simply download every image that is on the currently open webpage and store them in a zip folder [7]. The issue with this method is that the Louvre website only allows one hundred images per page and so it will have to be repeated to obtain the entire collection. It is however still considerably faster than downloading each individual image, allows a common name to be selected for all images and can filter out images that may be too small for the algorithm. If the need arises this method can also be used to obtain images for the training and validation datasets but will be more manual.

4.3. Image preparation

Machine learning algorithms require images to be of the same size and colour type when learning in order to keep data consistency. All the images above will therefore have to be converted to specific dimensions and the choice can be made to make the images grayscale which will improve processing times and can increase accuracy [8]. Both of these processes can be completed by a few lines of MATLAB code and either saved into a separate file and then loaded into the ML algorithm or converted as an initial step of the algorithm. A further step that has to be considered is that the images obtained from google may not all be desirable to train the algorithm with and may manually have to be removed from the datasets.

5. IMAGE CLASSIFICATION

ResNet has been established as one of the most effective image classification ML models due its "very deep" architecture as per the ImageNet dataset [9]. In comparison to other deep architectures such as AlexNet, VGG, etc. which simply stack additional layers, ResNet uses a *residual learning* framework instead. This addition of layers provides significant improvement in image classification while minimizing the degradation in testing error, prevalent in other architectures when increasing layers. This degradation is caused by the vanishing gradient problem [10] which occurs when learning a ML model using Gradient Descent and Backpropagation. During the backpropagation process, derivatives of every single layer are multiplied from the final to the initial layer. When trying to increase the number of layers, this process can cause the gradient to be so small that it has minimal effect on the weights and biases of initial few layers.

The backbone of the residual learning framework is known as the shortcut/skip connection. As illustrated in Figure 1, this connection is done by an element-wise addition of the input to the output of a layer. This helps mitigate the problem because it adds the input directly to the output resulting in a larger derivative.

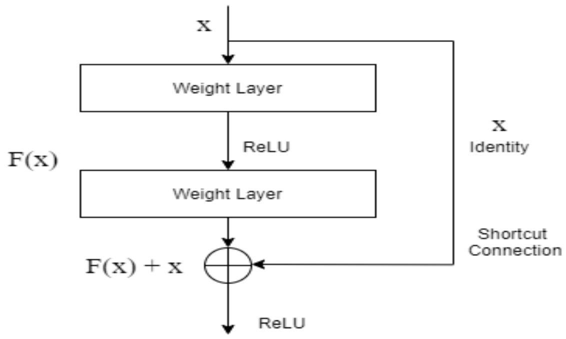


Figure 4: Shortcut connection used in Residual Learning

Initially, ResNet18 will be chosen for the task since a smaller depth offers lower complexity and fewer parameters (i.e. weights and biases) need to be trained. In the case that ResNet18 is not effective, the depth of the network can be increased to 34, 50 and so on until the required performance is reached.

Additionally, MATLAB also provides pre-trained ResNet models of depth 18, 50 and 101 which can be used for comparison and can be used for transfer learning to further optimize the model.

6. NEURAL NETWORK ARCHITECTURE

The convolutional neural network is a machine learning architecture that is based on a self-identification model. It is created from a collection of layers each of which modify the image in order to view the data from a different perspective and learn deeper features about the image. CNN's often have repetitive layer blocks that repeat a few key layers but have differing parameters in every layer. By changing the parameters in a layer, it will alter what information the neural network sees when training and so multiple blocks of these layers are employed to provide as many identifiers to an image as possible [2,11].

The three primary layers used in a CNN are the convolution, pooling and the fully connected layers. In order to implement the intended architecture these layers have to be fully understood. Figure 2 below presents the data representation of an image as well as key concepts associated with these layers.

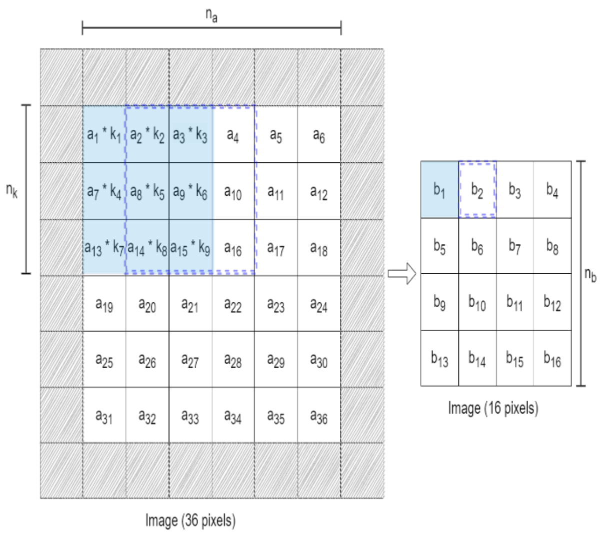


Figure 5: Model of a convolution layer filter being applied to an image.

6.1. Convolution layer

A convolution layer is the core of any CNN and is responsible for changing the weights of an image in order to determine a particular trend the image may exhibit. This is achieved by applying a $n_k \times n_k$ filter/kernel of values k_i over the image that will perform a convolution operation to each pixel. In figure 2 above the weight of an image's pixel is denoted by a_j, when the filter passes over an area each pixel undergoes the operation $a_j * k_i$. Every value in view of the filter is summed together resulting in the value b_1 in the right image. The blue dashed line represents the next position of the filter in order to obtain b_2 [2].

A filter is structured to look for a particular condition such as the horizontal and vertical edges but can also determine factors like luminosity. As can be seen in figure 2 above when applying the convolution layer the image has a decrease in size, equation 1 below represents this relationship based on the changeable parameters in the layer [2].

$$n_b = \frac{n_a + 2p - n_k}{s} + 1 \tag{1}$$

Where n_b is the new image size, n_a is the old size, n_k is the size of the filter applied, s is known as the stride which is how many rows or columns moved each time the filter is applied and p is known as padding. Padding is shown by the hatched border of the image in figure 2 and allows the programmer to maintain/increase an image size if the convolutional operation may not be beneficial with the current dimensions [2].

6.2. Pooling layer

A pooling layer is used to reduce the size of the image which aids in reducing computational load as well as preventing overfitting. It is implemented in a similar way to the convolutional layer in which a set of pixels will be observed by a filter type region. The difference lies computationally in that instead of performing a convolution operation, the entire area will just be changed to a single pixel either representing the max value (max pooling) or average value (mean pooling). Additionally, the filter does not move but rather encompasses every region of the original image that it can fit in without overlapping any other filters region [11]. With the example in figure 1 if a 3x3 filter is applied to a 6x6 image a pooling layer will then convert it to a 2x2 image.

6.3. Fully connected layer

The fully connected layer is used as one of the final layers in a convolutional neural network and acts as a collection of data from the previous layers. This data is passed into a feed forward neural network which allows predictions to be made on what features of the image most closely match a previous classifier [12].

7. TESTING METHODOLOGY

7.1. Validation

Due to the iterative nature of training a ML model, various factors need to be evaluated so that the model's architecture and hyperparameters can be optimized further. After each training stage, the model is evaluated using the validation dataset to generate a confusion matrix (error matrix) and precision-recall [13]. The confusion matrix is used to distinguish which classes/objects the algorithm is incorrectly classifying i.e. it identifies the true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN). Using these metrics the Precision can be calculated using (2) which identifies the portion of positives classifications that were correct [14].

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Similarly, Recall is calculated using (3) and identifies the portion of actual positives which were classified correctly.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Using these metrics, hyperparameters are tuned followed by the training and validation stages until the model is suitable.

7.2. Testing

After a suitable model is achieved, the model will be evaluated using the Testing dataset which is comprised of three sub-datasets spanning a century before Benjamin Franklin's arrival (1676 - 1775), during his time in France (1776 - 1785) and a century after his departure (1786 - 1886). To analyse his affect on French culture, the number of occurrences of lighting in paintings is compared between these three time periods.

8. PROJECT MANAGEMENT

8.1. Cost and Resources

No cost is expected to be incurred to ensure completion of the project as the group members have access to two laptops with sufficient storage and CPUs that are capable of running machine learning models. However, access to devices with GPUs and/or SSD storage would prove beneficial in providing increases efficiency and better overall performance for the model.

8.2. Risks and Mitigation

Several risks could derail the project and thus must be handled to ensure successful completion of the project. Insufficient training data poses a significant risk to the project since CNNs require large datasets for good accuracy. This can be mitigated by using data augmentation techniques such as translation, cropping and horizontal flipping to increase the dataset [15]. This also provides the benefit of regularization which prevents overfitting. Overfitting data is when the model is too closely coupled to the training data, thus it fails to fit new data.

The current socio-political climate as well as Covid-19 poses another risk which could result in lockdown or restrictions preventing access to campus. This means that the project will need to be completed remotely, at home. To mitigate this risk, each member will maintain a copy of the datasets on their device and the project will be stored externally on GitHub. This will ensure that both group members have access to the project remotely and will also prevent data loss. Additionally, progress meetings can be conducted on MS Teams or other platforms.

8.3. Timeline

The project will span six weeks starting on the 13th of September 2021. Since machine learning is an iterative process, not many tasks can be completed in parallel when the model is being trained or hyperparameters tuned thus the timeline of the project is very linear. A detailed project schedule and division of work can be found in Appendix A and consists of four stages: the *setup*, *initial* execution, *primary* execution and *extended* execution. Each stage includes testing using the Testing dataset to account for any delays that may occur during the process. This will ensure even if all stages are not completed, usable results are still available.

8.3.1. Setup

One week is allocated to data aggregation and preparation which includes gathering data for all the training sets, cleaning the data and augmenting it if the need arises. In addition, the software setup required should be completed during this stage.

8.3.2. Initial Execution

The initial execution stage is allocated 10 days as it includes the first implementation of the ML model. The aim of this stage is to produce a functioning ML model which is capable of providing results for the project. The results need not be accurate but are a minimum requirement for the project.

8.3.3. Primary Execution

The primary execution stage is allocated one week. The aim of this stage is to ensure that the primary objective of the project is achieved, thus the ML model should be able to identify lightning at least 60% of the time.

8.3.4. Extended Execution

The extended execution stage is allocated 6 days to provide a slack time in case delays occur. Additionally, the aim of this stage is to build on the work of the previous two stages and further research ways to optimize the algorithm to increase performance.

9. CONCLUSION

To meet the required project specifications a convolutional neural network using the ResNet18 architecture will be implemented because it is able to maintain and even increase its performance with an increase in layers. The datasets are obtained using python scripts for the training and validation and a Google Chrome extension is employed for the testing dataset. A project schedule is proposed to ensure that the work is completed within the 6-week timeline and the risks, costs and resources are mentioned as well.

10. REFERENCES

[1] "Deep Learning ToolboxDocumentation", *Mathworks.com*, 2021. [Online]. Available: https://www.mathworks.com/help/deeplearning/. [Accessed: 22- Jul- 2021].

[2] U. Michelucci, *Advanced Applied Deep Learning*. Berkeley, CA: Apress, 2019, pp. 79-104.

[3] E. J. Crowley, “Visual Recognition in Art using Machine Learning,” in PhD Thesis, University of Oxford, 2016.

[4] W. Zhao, D. Zhou, X. Qiu and W. Jiang, “Compare the performance of the models in art classification,” PLoS ONE, vol. 16, no. 3, 2021.

[5] B. Saleh and A. Elgammal, “Large-scale Classification of Fine-Art Paintings: Learning The Right Metric on The Right Feature,” arXiv:1505.00855, vol. 1, 2015.

[6] C. Zita, "How to Download Google Images, Using Python (2021)", gitconnected, 2021. [Online]. Available: https://levelup.gitconnected.com/how-to-download-google-images-using-python-2021-82e69c637d59. [Accessed: 26- Jul- 2021].

[7] "Download All Images", chrome web store, 2021. [Online]. Available: https://chrome.google.com/webstore/detail/download-all-images/ifipmflagepipjokmbdecpmjbibjnakm?hl=en. [Accessed: 25- Jul- 2021].

[8] C. Kanan and G. W.Cottrell, "Color-to-Grayscale: Does the Method Matter in Image Recognition?", *PLoS ONE*, vol. 7, no. 1, 2012. Available: https://journals.plos.org/plosone/article?id= 10.1371/journal.pone.0029740. [Accessed 28 July 2021].

[9] K. He, X. Zhang, S. Ren and S. Jian, “Deep Residual Learning for Image Recognition,” in In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[10] C.-F. Wang, “The Vanishing Gradient Problem,” Towards data science, 8 January 2019. [Online]. Available: https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484. [Accessed 31 July 2021].

[11] P. Kim, *Matlab deep learning*. New York: Springer, 2017, pp. 121-131.

[12] C. Aggarwal, *Neural Networks and Deep Learning*. New York: Springer, 2018, pp. 327-328.

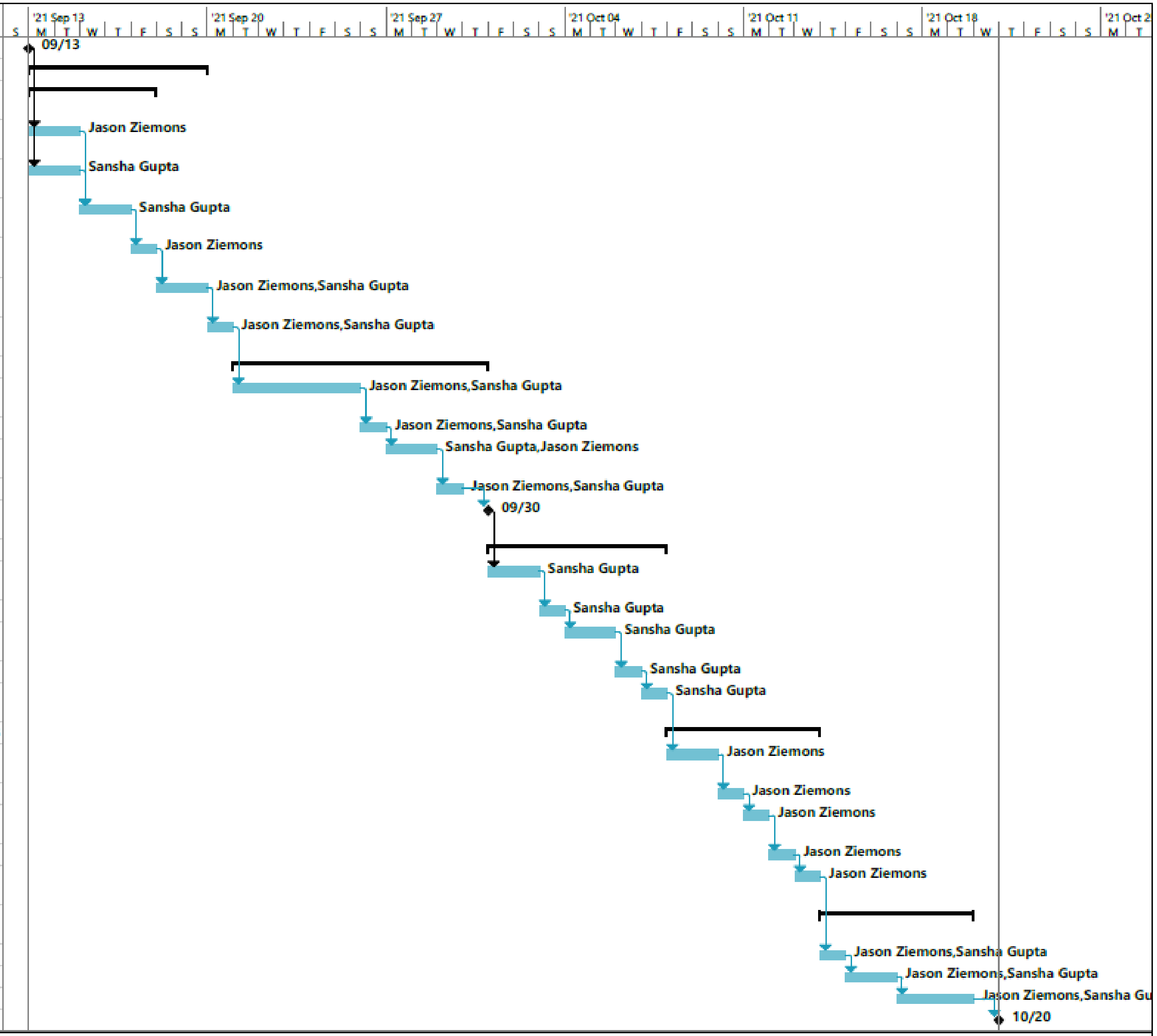
[13] R. Alake, "10 Stages Of A Machine Learning Project In 2020 (And Where You Fit)", *towards data science*, 2020. [Online]. Available: https://towardsdatascience.com/10-stages-of-a-machine-learning-project-in-2020-and-where-you-fit-cb73ad4726cb. [Accessed: 30- Jul- 2021].

[14] "Classification: Precision and Recall | Machine Learning Crash Course", *Google Developers*. [Online]. Available: https://developers.google.com/machine-learning/crash-course/classification/precision-and-recall. [Accessed: 30- Jul- 2021].

[15] D. Ho, E. Liang, I. Stoica, P. Abbeel and X. Chen, "Population Based Augmentation: Efficient Learning of Augmentation Policy Schedules", vol. 1, 2019. Available: https://arxiv.org/pdf/1905.05393.pdf. [Accessed 1 August 2021].

Appendix A

ID		Task Mode	Task Name	Duration	Start	Finish
1			Start	0 days	Mon 21/09/13	Mon 21/09/13
2			**Setup**	**7 days**	**Mon 21/09/13**	**Sun 21/09/19**
3			**Generate training dataset**	**5 days**	**Mon 21/09/13**	**Fri 21/09/17**
4			Data gathering (Training)	2 days	Mon 21/09/13	Tue 21/09/14
5			Data gathering (Validation +	2 days	Mon 21/09/13	Tue 21/09/14
6			Data cleaning and	2 days	Wed 21/09/15	Thu 21/09/16
7			Data preparation	1 day	Fri 21/09/17	Fri 21/09/17
8			Environment setup (MATLAB)	2 days	Sat 21/09/18	Sun 21/09/19
9			Determine appropriate model	1 day	Mon 21/09/20	Mon 21/09/20
10			**Initial Execution**	**10 days**	**Tue 21/09/21**	**Thu 21/09/30**
11			Implementation of chosen model	5 days	Tue 21/09/21	Sat 21/09/25
12			Evaluation	1 day	Sun 21/09/26	Sun 21/09/26
13			Hyperparameter tuning	2 days	Mon 21/09/27	Tue 21/09/28
14			Evaluation	1 day	Wed 21/09/29	Wed 21/09/29
15			Test using Louvre dataset	1 day	Thu 21/09/30	Thu 21/09/30
16			**Primary Execution**	**7 days**	**Fri 21/10/01**	**Thu 21/10/07**
17			Hyperparameter tuning	2 days	Fri 21/10/01	Sat 21/10/02
18			Evaluation	1 day	Sun 21/10/03	Sun 21/10/03
19			Hyperparameter tuning	2 days	Mon 21/10/04	Tue 21/10/05
20			Evaluation	1 day	Wed 21/10/06	Wed 21/10/06
21			Test using Louvre dataset	1 day	Thu 21/10/07	Thu 21/10/07
22			**Extended Execution**	**6 days**	**Fri 21/10/08**	**Wed 21/10/13**
23			Hyperparameter tuning	2 days	Fri 21/10/08	Sat 21/10/09
24			Evaluation	1 day	Sun 21/10/10	Sun 21/10/10
25			Hyperparameter tuning	1 day	Mon 21/10/11	Mon 21/10/11
26			Evaluation	1 day	Tue 21/10/12	Tue 21/10/12
27			Test using Louvre dataset	1 day	Wed 21/10/13	Wed 21/10/13
28			**Open Day Presentation**	**6 days**	**Thu 21/10/14**	**Tue 21/10/19**
29			Plan Presentation	1 day	Thu 21/10/14	Thu 21/10/14
30			Film video	2 days	Fri 21/10/15	Sat 21/10/16
31			Edit video	3 days	Sun 21/10/17	Tue 21/10/19
32			Demonstration	0 days	Wed 21/10/20	Wed 21/10/20



APPENDIX D – Report supporting material

Table 1: Detailed table of results for determining ideal hyper-parameters (same table split across 2 images)

Run	CNN used	Mini-Batch size	Epochs	Initial learning rate	Lightning images	Number of anchors	Optimizer	Feature layer	MeanIOU	MAP
1	Resnet 18	32	20	0.001	3031	10	adam	res5b_relu	0.6851	0.58
2	Resnet 18	32	20	0.001	3031	10	rmsprop	res5b_relu	0.6851	0.48
3	Resnet 18	32	20	0.001	3031	10	sgdm	res5b_relu	0.6851	0.41
4	Resnet 18	32	20	0.001	3031	1	adam	res5b_relu	0.4036	0.58
5	Resnet 18	16	5	0.001	3031	1	adam	res5b_relu	0.4036	0.47
6	Resnet 18	32	5	0.001	3031	1	adam	res5b_relu	0.4036	0.49
7	Resnet 18	64	5	0.001	3031	1	adam	res5b_relu	0.4036	0.49
8	Resnet 18	32	20	0.0005	3031	3	adam	res5b_relu	0.5989	0.58
9	Resnet 18	32	20	0.0001	3031	1	adam	res5b_relu	0.4036	0.55
10	Resnet 18	32	10	0.001	3031	6	adam	res5b_relu	0.6805	0.57
11	Resnet 18	32	20	Piecewise (0.001,0.5,5Epochs)	3031	1	adam	res5b_relu	0.4036	0.60
12	Resnet 18	64	20	Piecewise (0.001,0.5,5Epochs)	3031	1	adam	res5b_relu	0.4036	0.57
13	Resnet 18	32	20	Piecewise (0.01,0.5,5Epochs)	3031	1	adam	res5b_relu	0.4036	0.27
14	Resnet 18	32	5	0.001	3031	1	adam	res2b_relu	0.4036	0.0
15	Resnet 18	32	5	0.001	3031	1	adam	res3b_relu	0.4036	0.0
16	Resnet 18	32	5	0.001	3031	1	adam	res4b_relu	0.4036	0.54
17	Resnet 18	32	5	0.001	3031	1	adam	res5b_relu	0.4036	0.51
18	Darknet-19	32	20	0.001	3031	1	adam	res5b_relu	0.4036	0.55
19	Resnet 18	32	20	0.001	3031	1	adam	res4b_relu	0.4036	0.60

Run	Louvre images identified with 50% tolerance	122_500 images dataset images found with 50% tolerance	Correct lightning images found (500 images)	Percentage of Louvre images correct	Percentage of images found (122)
1	197	95	57	0,071428571	0,448818898
2	816	168	58	0,071428571	0,456692913
3	44	26	23	0	0,181102362
4	570	128	65	0,357142857	0,511811024
5	397	98	40	0,071428571	0,31496063
6	221	57	38	0,142857143	0,299212598
7	334	94	52	0	0,409448819
8	865	207	63	0,142857143	0,496062992
9	270	90	52	0,071428571	0,409448819
10	570	151	63	0,142857143	0,496062992
11	396	129	60	0,214285714	0,472440945
12	298	109	64	0,214285714	0,503937008
13	146	49	5	0	0,039370079
14	0			0	0
15	0			0	0
16	781				0
17	270				0
18	301	110	56	0,285714286	0,440944882
19	393	129	66		0,519685039

APPENDIX E – Meeting minutes

MEETING MINUTES

Meeting Information

Week: 1
Date: 17th September 2021
Begin Time: 9:00 am
End Time: 9:26 am

Location: Microsoft Teams
Chairperson: Prof. Estelle Trengove
Secretary: Daniel Katz

Meeting Attendees

Guest: None
Supervisor: Prof. Estelle Trengove
Students: Jason Ziemons, Sansha Gupta, Daniel Katz, Yair Bisnowaty

Opening Remark

Prof. Trengove welcomes the students to the first official meeting of the prescribed 6-week Laboratory Project.

Apologies

None

Agenda

The agenda is to commence the project and discuss the current progress of each group, along with any problems encountered and goals for the upcoming week.

Main Motion

1. Jason and Sansha's project (Group 54):

- The group has been able to resource information pertaining to AI courses in MATLAB. The courses deal with basic concepts used in machine learning, such as data collection, dataset partitioning, and basic image processing.
- The group encountered issues with copyright infringements with downloaded images.
- The group also raises concern of the very long processing times it takes to train a neural network.
- A possible solution to this problem is to consult with either Dr Ken Nixon or Mr Scott Hazzlehurst to allow access to the Jaguar cluster at Wits, in the coming week. The cluster has GPUs that can drastically speed up processing time.

2. Yair and Daniel's project (Group 9):

- The group has identified a python library, MediaPipe, that can detect human landmarks in videos.
- A brute-force algorithm analysing the angle between the hip, knee, and ankle, as well as feet positions, has been implemented to detect when a footstep occurs. The students have experienced some success with this regard, however the data requires filtering and cleaning in order to be reliable.
- The group has also found the python libraries PyDub and MoviePy which can be used to create audio tracks and combine video and audio files.
- The group plans to train an AI, in the coming week, to automatically detect footsteps without brute-force methods.
- The group asks the supervisor whether there are any experts in the field of deep learning that can assist in the training procedure.
- Prof Estelle provides details about Vered Aharonson.

3. Additional Discussions:

- Each group familiarised the other group with their respective project specifications.
- Jason and Sansha are required to build a model that can detect lightning in images/pieces of art.
- The goal is to determine in some way whether there is correlation between his popularity and the presence of lightning in artworks of the time.
- Daniel and Yair explain the project outline of matching footsteps to videos. The goal of the project is to build a model that can detect footsteps in a silent film and add a synchronized footstep soundtrack to the original video.

Meeting Adjournment

The meeting is concluded with both groups stating the need to do further research in the required fields to be able to tackle the Laboratory Project problem.

MEETING MINUTES

Meeting Information

Week: 2
Date: 23rd September 2021
Begin Time: 9:28 am
End Time: 9:54 am
Location: Microsoft Teams
Chairperson: Prof. Estelle Trengove
Secretary: Jason Ziemons

Meeting Attendees

Guest: None
Supervisor: Prof. Estelle Trengove
Students: Jason Ziemons, Sansha Gupta, Daniel Katz, Yair Bisnowaty

Opening Remark

Everybody made their introductions and we moved onto the meeting agenda for the week.

Apologies

None

Agenda

The aim of the meeting is to determine the progress that has been made thus far and discuss future plans as well as any complications that have been experienced.

Main Motion

1. Jason and Sansha's project (Group 54):
   - *Progress report*: haven't done any training yet due to issues with Linux and MATLAB so we spent time learning Python and trying to integrate MATLAB and Linux with Nixon's cluster.
   - *Plans for coming week*: Will hopefully be able to start training the model image classifier on the cluster using MATLAB otherwise will have made the shift to Python and fully implement working code.

2. Yair and Daniel's project (Group 9):
   - *Progress report:* built an algorithm to identify footsteps, stride and length and it is workings decently (brute force algorithm), managed to collect some data with different types of footsteps but had some issues with the library they were using, managed to pick up the left foot and right foot strides
   - *Unexpected obstacles:* it's only taking a certain number of frames and it needs to be a fixed variable, so they are adjusting to fix that, planning to add more data due to concerns of achieving a 100% accuracy score.
   - *Plans for coming week:* finalise and synthesize with audio and then look at exploring different surfaces.

3. Additional discussions:
   - Would be beneficial to write a small section in our reports of the things we considered and rejected for a specific reason such as our MATLAB issues.
   - Is valid to refer to using an iterative process and we should discuss our process of solving the problem and not only to discuss the final solution.
   - Daniel asked if you need ethics clearance for recording people walking? : Estelle suggests you should probably record each other walking or you may be able to just ask the person for specific permission and create a form template that they have signed to be placed in the appendix.
   - Yair asked us how we are tracking the moving lightning/video, but we are doing static images, Daniel recommended a labelling package in an API in Python and that we can message him for any more info.
   - Yair suggested an in-person meeting and to possibly include Adam Pantonowitz to help us discuss more about machine learning for our next week's meeting, possibly Turgay Celik or Oharason as well.

Meeting Adjournment

It was concluded that Adam Pantanowitz may be able to join us in the following meetings in order to aid with the more technical aspects of machine learning.

MEETING MINUTES

Meeting Information

Week: 3
Date: 1st October 2021
Begin Time: 9:00 am
End Time: 9:50 am
Location: Microsoft Teams
Chairperson: Adam Pantanowitz
Secretary: Sansha Gupta

Meeting Attendees

Guest: Adam Pantanowitz
Students: Jason Ziemons, Sansha Gupta, Daniel Katz, Yair Bisnowaty

Opening Remark

Prof. Adam introduced himself to the students and the chair welcomed everyone to the meeting and introduced the agenda.

Apologies

Prof Trengove was not able to attend the meeting.

Agenda

1. Introduce topics to Adam
2. Update on previous week's progress
3. Any challenges and issues faced?
4. Goals for the upcoming week

Main Motion

1. Jason and Sansha's project (Group 54):

Introduced topic and current progress of the classifier model to Prof Adam. The group has managed to train a CNN classifier to detect lightning, however due to the variations in photos in the test dataset, they are unsure how to proceed. Suggestions by Prof Adam on next steps are as follows:

- Introduce minimum threshold that distinguishes between lightning and non-lightning paintings
- Need to pre-process data to transform it to correct perspective on the paintings of the Louvre dataset.
- Merge filtered training photos and normal training photos into one large dataset.
- Possible use of object detection to find the sky in paintings.
    - Just choose the top half of the image as a first variation on this.
    - Can also use an ML model to accomplish this.
- Possibly use grayscale images to diminish variability in coloured depictions of lightning.
- Consider traditional machine learning approaches and use HOG or Gabor filters for feature extraction.

2. Yair and Daniel's project (Group 9):

Introduced topic and current progress on showing proof of concept for foot sounds to Prof Adam. The group is utilizing a pose estimation tool for landmarks on the body and detecting the exact position of the feet. They are currently attempting to detect the type of surface being walked on using an object detection system. The SSD method has failed. Prof Adams suggestions for further steps include:

- Take foot position and sample a little further down to get floor type. Can use a random forest classifier or easier to use CNN perhaps. 40 samples from each floor type.
- Using an image classifier instead of an object detector for this task.
- Also discussed the process of detecting the type of shoe worn.

Meeting Adjournment

Thanked Prof Adam for joining and the suggestions given. Prof Adam will be joining the following meetings to discuss progress and how the suggestions went.

MEETING MINUTES

Meeting Information

Week: 4
Date: 8th October 2021
Begin Time: 9:00 am
End Time: 9:43 am
Location: Microsoft Teams
Chairperson: Prof. Estelle Trengove
Secretary: Yair Bisnowaty

Meeting Attendees

Guest: Adam Pantanowitz
Supervisor: Prof. Estelle Trengove
Students: Jason Ziemons, Sansha Gupta, Daniel Katz, Yair Bisnowaty

Opening Remark

Prof. Estelle welcomes everyone to the meeting and thanks Adam for chairing the meeting last week.

Apologies

None

Agenda

The objective of the meeting is to discuss the progress of the week, ask any questions and advice, and express any future goals to be achieved in the following week.

Main Motion

1. Jason and Sansha's project (Group 54):
   - Jason explained their progress and mentioned their HOG extraction issues and that they were having too many false negatives in their project. Jason and Sansha explained that there are issues with paintings that contain a lot of glares.
   - Adam recommended using Gabor filters for their classification.
   - Jason explains that they are using the entire image for training the model.
   - Adam suggests segmenting the sky from the rest of the image to isolate a region of interest.

2. Yair and Daniel's project (Group 9):
   - Yair explains the lack of data available.
   - Adam suggests using data augmentation and use actual generated data from the videos.
   - Daniel shows everyone the progress and demonstrates how Mediapipe detects humans, displaying landmarks.
   - Adam suggests using binary classification with footstep-no footstep instead of left and right footstep detection.
   - Yair asks if it's legal and ethical to record people walking in public.
   - Estelle and Adam both state its legal and ethical if the data is not published and used only for training of the model.
   - Dan explains his progress in his surface classification and the difficulty in image processing.
   - Adam suggests using single colour transformations and/or grey scaling.
   - Estelle requests Daniel and Yair to present their project to the Dean on the 8th of November 2021.

3. Additional Discussions:
   - Adam suggested keeping records of all engineering process.

Meeting Adjournment

The meeting concluded with Prof. Estelle thanking Adam for assisting the students and reiterating that the project has surpassed the half-way mark.

MEETING MINUTES

Meeting Information

Week: 5
Date: 15th October 2021
Begin Time: 9:04 am
End Time: 9:38 am
Location: Microsoft Teams
Chairperson: Daniel Katz
Secretary: Daniel Katz

Meeting Attendees

Guest: Adam Pantanowitz
Supervisor: Prof. Estelle Trengove
Students: Jason Ziemons, Sansha Gupta, Daniel Katz, Yair Bisnowaty

Opening Remark

Prof. Estelle Trengove indicates that this is the final meeting before the Project Poster submission and Virtual Open Day, thus entering the final leg of the project.

Apologies

None

Agenda

Discuss the progress of each project to ensure students are at the necessary stage for open day and poster submissions. Adam Pantanowitz is present to lend assistance with any lingering problems the students are facing. Report writing and the final presentations are to be discussed.

Main Motion

1. Jason and Sansha's project (Group 54):
    - The group has resorted to using an object detection model to distinguish clouds and lightning as previous models were constantly confusing them. The model is struggling with a lot of false positives due to thresholding in the pre-processing layer of the model. The model is currently achieving a 70% validation accuracy.
    - Adam suggests that image segmentation should be used to separate sky and land and the sky region used in an image classification model. Lowering the threshold to allow for more images to be inputted into the classifier.
    - Sansha has questions about Random Classifier

2. Yair and Daniel's project (Group 9):
    - Yair expressing difficulty with shoe classification using Convolutional Neural Network. Yair has trained his model using the existing MNIST dataset of shoe classes which have been pre-processed and look very different from the images he has captured from his own program. Yair has tried to process the input images using computer vision techniques such as thresholding, Gray scaling, and other filters in an attempt to look like the MNIST ones. The model is still failing to classify shoes correctly.
    - Adam explains the reason the for the classification failing is due to a conflict between training and testing data. Adam suggests training the CNN with the extracted images from Yair's data. A further suggestion is given that the shoe be extracted when flat to reduce shadows and to only extract shoes and not floor.
    - Daniel has discovered another pose classification model that uses a kNN structure using only two frames to define a footstep using Mediapipe.
    - Adam states that it is too late in the project to change course. The students must focus on getting what they have already achieved to work.

4. Additional Discussions:
    - Prof Estelle explains how to construct the poster in such a way that it can be understood by non-technical people. Putting lots of images and few words is a useful strategy to building a good poster.
    - Prof Estelle also adds advice for the upcoming presentation. The presentation must be finished in good time and not run out of time. The slides used must be neat and have lots of graphs and results.

Meeting Adjournment

This was the last official meeting between supervisor and students. Plans were made to organise private meetings to be scheduled using the group's WhatsApp group, should the need arise. This is done for any questions pertaining to report writing and the presentation.