Queensland University of Technology

MXB362:
ADVANCED VISUALISATION AND DATA SCIENCE

Final Report (40%)

Semester 2, 2023

PROJECT:

Interactive Visualisation of the
Segmentation Model Training Progress

Student Name: **Thanh Long (Oliver) Vu**

Student No: **N11177071**

TABLE OF CONTENTS

1. Executive Summary

In the ever-evolving domain of machine learning, learners and students often grapple with understanding the intricate model training progress and the evolution of a model. The prevailing solutions predominantly offer static visualisations, lacking in dynamic information and interaction, which accentuates the gap between complex training processes and intuitive understanding.

To bridge this gap, our primary objective was to develop a dynamic and interactive visualisation that offers real-time insights into model training to turn the often confusing training phase into a clear visual journey. For this endeavour, we chose the Segmentation Problem, utilising the Oxford Pets Dataset as our foundational study. This dataset, emphasising the segmentation of animals from their backgrounds, served as our primary case study. The core of our visualisation centres on showing how the model, underpinned by the MobileNetV3Small base-model, evolves over a span of 30 epochs.

The primary outcome of our project is a web-based unified dashboard. It features a heatmap (depicting a Confusion matrix), a line graph showcasing test accuracy, and a side-by-side comparison of the original image and its predicted mask.This dashboard not only shows model performance results but also simulates the model training progress in real-time, leveraging a progress bar to provide users with an immersive experience, thereby filling the existing void left by static visualisations.

Data used for the visualisations were meticulously collected during the model training phase. This data set encompasses Testing accuracy, Confusion Matrix details, and predicted masks for each model iteration across 30 epochs. After data collection, Matplotlib was employed for visualisation, while Streamlit was chosen to craft the web-based dashboard. This allowed for a fluid simulation of the model training process, complete with interactive buttons and a progress bar. Streamlit's row and column structure rendered the final dashboard both effective and congruent with our prototype.

Our visualisation's design and architecture was based on multiple visualisation effectiveness theories and principles. Every facet, from visualisation type selection to elemental design, was deliberately chosen to optimise effectiveness. Even our dashboard's design was meticulously curated, with research-backed theories steering our choices to maximise clarity and user engagement.

In conclusion, this project successfully achieved the proposed objective of crafting an effective visualisation that brings unprecedented clarity to the model training process. The project, now live, beckons a wider audience to experience its offerings. Looking ahead, there's a palpable excitement for the future, as we envisage expanding the project's horizons, incorporating a plethora of features to continually enrich the learning experience for our target audience.

2. Project Description

2.1. Project Objective

Artificial Intelligence (AI) and Machine Learning (ML) have surged in popularity due to their potential to resolve a myriad of real-world problems, consequently attracting an increasing number of students and enthusiasts.

Despite this growing interest, Machine Learning and Artificial Intelligence are notoriously complex - often perceived as a ‘black box’ - a system whose workings are mysterious and unclear - for new learners, particularly during the critical stages of training and fine-tuning models. This complexity poses significant hurdles for newcomers, who may find it challenging to grasp the nuances of model progression through each epoch, leading to potential misunderstandings and roadblocks in their learning journey.

As a result, our project seeks to bring clarity to the model training progress by developing an interactive visualisation that provides real-time, intuitive insights into the model's learning progression. This tool is designed not only to make the learning process more accessible and engaging for students and beginners but also to empower them with the clarity needed to make informed decisions in model training and evaluation.

By bridging this knowledge gap, our initiative is poised to significantly benefit our primary target audience: students and newcomers in the fields of data science, information technology, and computer science, who are grasping the fundamentals of AI and ML. Our ultimate goal is to transform the ambiguous training phase into a comprehensible visual journey, making these complex areas more approachable and understandable with the aid of effective visualisation.

2.2. Background research

The landscape of tools available for visualising machine learning model training is diverse, yet each tool comes with its own set of capabilities and limitations.

According to the 2018 Python Developers Survey by JetBrains, Matplotlib enjoys widespread usage with 46% of respondents leveraging it, likely due to its versatility in generating a multitude of plots essential for data analysis and interpretation. Keras, used by 15% of the participants, is also notable, particularly for its seamless integration with TensorFlow, facilitating neural network experimentation.

These libraries are indispensable when it comes to plotting fundamental graphs such as line graphs for tracking accuracy and loss metrics over epochs, bar plots for visualising feature importance, or scatter plots for tasks like clustering which require data point comparisons. Additionally, Seaborn, an extension of Matplotlib, allows for more advanced statistical visualisations, for instance, heatmaps that are effective for illustrating confusion matrices, thus providing more nuanced insights into model performance.

However, despite their utility, these tools share a significant limitation: they primarily support static visualisations generated post model training, creating a gap in real-time feedback during the model training process. Real-time visualisations, on the other hand, allow for a clear understanding of how a model evolves and different factors such as different base model, epochs affect the model progression.

This deficiency, therefore, suggests an opportunity for tools that can bridge this gap, offering rich, real-time visual insights that can transform the learning experience and efficacy for individuals delving into the world of machine learning

2.3. Visualisation problem context

2.3.1. Segmentation Problem

In machine learning, segmentation involves partitioning a dataset into distinct groups or segments, often to identify patterns or categorise the data. This technique is especially pertinent in image processing, where image segmentation is employed to divide an image into multiple parts. This division is typically based on certain criteria, such as colour, intensity, or object type, with each pixel in the image being assigned to a particular segment or category.

Image segmentation plays a crucial role in various practical applications, from object detection in autonomous vehicles to medical imaging for disease diagnosis. It's particularly important in fields requiring precise image analysis, such as facial recognition or wildlife tracking.

2.3.2. Application in the project

For this visualisation project, we've selected an image segmentation task aimed at distinguishing pets from the background in images. This task is not only relatable, given our everyday encounters with pets, but also challenging due to the variety in pet appearances and backgrounds. By focusing on this problem, we aim to demonstrate the intricacies of model progression in machine learning, highlighting how models learn and adapt to achieve successful segmentation. This visual representation of model training will provide learners with a clearer, more tangible understanding of machine learning concepts in action.

2.4. Data source

2.4.1 Dataset Description for Model Training

- **Oxford Pets Dataset Composition**: The Oxford Pets dataset, serving as the foundation for our model's training, boasts a diverse array of colour images, with each pet portrait teeming with rich textures, variances in fur patterns, and a spectrum of shapes. For model training coherence, these images have been resized to a 128x128 pixel resolution. This size was strategically chosen to balance computational demands with the preservation of meaningful visual details.

2.4.2 Visualisation Data Description

Our visualisation will lean heavily on the meta-data that emerges during the model training process, including:

- **Accuracy Over Epochs:** Presented in a dynamic line graph, this metric showcases the model's learning curve, reflecting its consistent growth in adeptness at the segmentation task. It serves as a visual testament to the model's evolving proficiency.

- **Segmentation Performance:** This section displays the predicted masks, elucidating the model's capacity to segment distinct image regions correctly. Comparing these masks with the actual segmentations allows for an intuitive grasp of the model's current prowess.
- **Confusion Matrix**: Beyond simple classification rates, this matrix offers a detailed breakdown of the model's performance. It provides quantified data on true positives, false positives, true negatives, and false negatives, portraying the model's precision and recall capabilities in relation to the dataset's complex attributes.

2.5. Visualisation Techniques

Given the nature of our data and the objective of providing a transparent window into the intricacies of model training, we will employ a combination of established and innovative visualisation techniques to effectively communicate the progression and performance of our model. Here's a breakdown:

Line graph: Specifically tailored for our project, line graph plays a crucial role in tracing the model's accuracy trajectory over the epochs. This graph illuminates the dynamic changes in accuracy, offering a clear visual representation of how the model's performance enhances or fluctuates with each successive epoch. By following this graphical trend, observers can effortlessly comprehend the model's continual development throughout the training phase.

Confusion Matrix (Heatmap): An indispensable tool in classification tasks, the Confusion Matrix offers a snapshot of the model's predictions against actual outcomes. By converting these results into a heatmap, users can quickly identify metrics like precision, recall, or F1 score.

Segmentation Performance Visualisation: This showcases the model's capability in distinguishing target objects (like animals) from the background. By juxtaposing the original image with the model's prediction, users gain a clear perspective of the model's accuracy in a real testing image.

A deeper dive and comprehensive discussion on visualisation types and techniques will be covered in subsequent sections of the report.

2.6. Visualisation Environments and Tools

This project employs a suite of streamlined tools and environments, each chosen for its efficiency and compatibility with our visualisation objectives.

Development Environment:
- Platform/IDE: We use VSCode for its robust Python support and Jupyter Notebook for its interactive computing environment.
- Operating System: Development is conducted on macOS, preferred for its stability and performance.

Programming Language:
- Python: Chosen for its simplicity, readability, and extensive libraries, Python is fundamental to all stages of our project.

Data Processing:
- Pandas: Utilised for efficient data manipulation and analysis.
- Numpy: Employed for optimal numerical operations and array processing.

Model Training:

- Tensorflow: Our go-to open-source library for model training and evaluation, prized for its comprehensive tools.
- Keras: Used for simplified model architecture design and training.

Visualisation:
- Matplotlib: Essential for creating diverse visualisations, turning complex data into understandable visuals.
- Streamlit: Enables us to build interactive web applications quickly, making our visualisations accessible and manipulable.

3. Results and Outputs

3.1. Algorithms

For the visualisation project, our primary intent was to develop a model with acceptable performance that vividly illustrates the nuances between different epochs. We did not prioritise achieving perfect accuracy or extending training durations; rather we desired a transparent model that could highlight the progression dynamics. To achieve this, our algorithm and model training strategy encompassed the following components:

Base Model: We employed the MobileNetV3Small as our base model. MobileNetV3Small is known for its lightweight architecture and efficiency, making it suitable for our segmentation project without compromising on computational power or speed.

Decoder Architecture: To process our segmented images, we used a decoder architecture inspired by U-Net which comprises upsampling, convolution, batch normalisation, and spatial dropout layers. U-Net is a renowned convolutional network designed for biomedical image segmentation. This architecture is especially favourable due to its capability to capture context and localise, essential attributes for our segmentation problem.

We adopted a two-stage training strategy:
- Initial Training: We initially trained our model for 20 epochs with a frozen base model. This step ensures that the weights from the pretrained MobileNetV3Small model remain unchanged and serve as a stable foundation.
- Fine-tuning: After the initial training, we fine-tuned our model for an additional 10 epochs, allowing the base model to further refine and optimise its weights based on our specific dataset.

Our model exhibited a commendable accuracy of 0.874 on the testing set, signifying its robustness and efficiency in the given segmentation task.

3.2. Data Processing & Manipulation

3.2.1. Data Collection

Before visualisation creation, data used for these visualisations is meticulously collected in advance during the model training process

The model is trained over 30 epochs, with data saved at the conclusion of each epoch. This allows for a retrospective examination of the model's learning progression at various intervals.

The model's performance is uniformly assessed against a consistent test set, ensuring that evaluation conditions remain consistent across epochs.

By following this approach, we streamline the visualisation process by removing prolonged training durations. This enhancement accelerates the visualisation process, providing users with quicker insights. Three main types of data are collected during the process:
- **Accuracy Data**: Reflects the model's accuracy across the 30 epochs on the test dataset.
- **Confusion Matrix Data**: Captures metrics such as True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN) over the 30 epochs. This data provides a holistic view of the model's predictive accuracy and areas that may require refinement.
- **Segmentation Frames**: These are periodic frames showing predicted results with the actual image, illustrating the model's segmentation abilities throughout its training phases.

The dataset presents a challenge with a diverse array of textures, splitting across multiple files. It requires further processing and manipulation before visualisation creation.

3.2.2. Data Processing & Manipulation

During the data processing and manipulation phase, several key steps were undertaken to ensure optimal visual representation. The model's training process resulted in individual CSV files for each of the 30 epochs. To streamline our workflow, we aggregated these files into a single consolidated dataset. Following the aggregation, it was crucial to check the unified dataset for any missing or mismatched records, ensuring data integrity and an accurate depiction of the model's progression.

After data processing and manipulation, we obtained a single CSV file containing both accuracy and confusion matrix data, along with a GIF for visualisation.

3.3. Visualisation implementation & Novel Techniques

3.3.1. Individual visualisation implementation

For the accuracy line graph and confusion matrix, we employed Python's popular plotting library, Matplotlib. This library facilitates the creation of a myriad of intricate visualisations with vast customization options.

Accuracy Line Graph:
For accuracy, we've set up a dynamic line graph that portrays the segmentation accuracy of the model across the training epochs.

Firstly, we initialise the plot with create_initial_plot(). This sets the foundational structure of our graph: its dimensions, the x-axis to represent epochs (from 1 to 30), and a y-axis that adjusts to the range of segmentation accuracy values in our dataset.

The graph's continuous update, to reflect performance for each epoch, is achieved through the update_plot() function. By fetching accuracy data corresponding to the current epoch from the consolidated CSV, this function ensures the plot showcases the real-time progression of the model's training accuracy.

Confusion Matrix:

For the confusion matrix, the primary aim was to visually encapsulate the model's prediction outcomes – True Positives, False Positives, True Negatives, and False Negatives – throughout the training epochs.

The base structure of the matrix is determined by the create_initial_confusion_matrix() function. It sets the matrix's dimensions, its colour scale, and other essential parameters such as the 'Positive' and 'Negative' categories for both predicted and actual outcomes.

The dynamism of the confusion matrix is maintained by the update_confusion_matrix() function. For each epoch, this function retrieves the relevant matrix values from the consolidated CSV, ensuring the displayed data remains current with the model's training phase.

Segmentation Performance Visualisation Implementation:
To provide a comprehensive and dynamic illustration of the model's segmentation performance across training epochs, we amalgamated the 30 segmentation frames into a GIF. The duration of each frame in the GIF was meticulously calibrated to align with the epoch simulation times, offering an authentic depiction of the model's evolving learning curve.

3.3.3. Web-based Dashboard

The dashboard is structured using Streamlit's layout capabilities, leveraging its columns and rows feature. This design approach allowed us to compartmentalise various elements, ensuring that the visualisations, buttons, and other controls are neatly organised within the dashboard.

A button has been added to the dashboard, serving as a trigger to start or restart the simulated training process. Complementing this, a progress bar is employed to represent the advancement through epochs. As the user initiates the training via the button, the progress bar begins filling up proportionally with each simulated epoch.

A coordinated update mechanism has been set up to ensure that as the progress bar advances, multiple visual elements of the dashboard get updated concurrently. Each visual element, like the accuracy plot and confusion matrix, is refreshed in synchrony with the simulated epoch times. This is achieved by aligning the duration of each frame update in the visual elements with the progression of the training simulation.

3.3.4. Novel Techniques

Training Simulation Instead of Real-time Training:
Instead of real-time model training, we employed a simulation based on pre-collected data. Real-time training, while authentic, posed challenges. Notably, completing one epoch would require around 2 minutes, with an additional 10 seconds to fit the model on the testing set. This waiting time, when projected over the entire training period, would have been considerably lengthy and would hamper the visualisation's overall impact.

By opting for a simulated training mechanism, we bypassed this extended training duration. This not only accelerated the visualisation process but also augmented the overall user experience.

Employing GIFs for Visualising Predicted Masks:
Rather than generating real-time predicted masks, we used GIFs to represent the model's segmentation performance across epochs. This method ensured smooth transitions and

synchronised perfectly with the 0.5-second epoch duration in our simulated training, eliminating any user-side waiting time.

3.4. Visualisation outputs

The final output of this project is an interactive, web-based dashboard. The complete visualisation can be viewed at: https://olivervu25-segmentationtrainingvisualisation-dashboard-kjvauo.streamlit.app/.

This dashboard has been structured to present users with a comprehensive understanding of the model's performance, comprising three primary visualisations:

- **Confusion Matrix**: Strategically positioned on the top-left, it provides an in-depth insight into the model's classification accuracy.
- **Test Accuracy Line Graph**: Found below the Confusion Matrix, this graphical representation plots the model's accuracy on the testing dataset over successive epochs.
- **Segmentation Performance Visualisation**: Placed on the bottom-right, this visualisation elucidates the model's proficiency in segmentation tasks.

Complementing these primary visualisations is the "Start Training" activation mechanism. When initiated, this function simulates the model's training process. As the simulation progresses, the accompanying progress bar updates to reflect the current epoch, while all three primary visualisations dynamically adjust to display the model's concurrent performance on the testing dataset.

It is important to note an informative cue introduced at epoch 20. This cue signifies the unfreezing of the MobileNetV3Small base model, serving to apprise users of significant transitions in the training methodology.

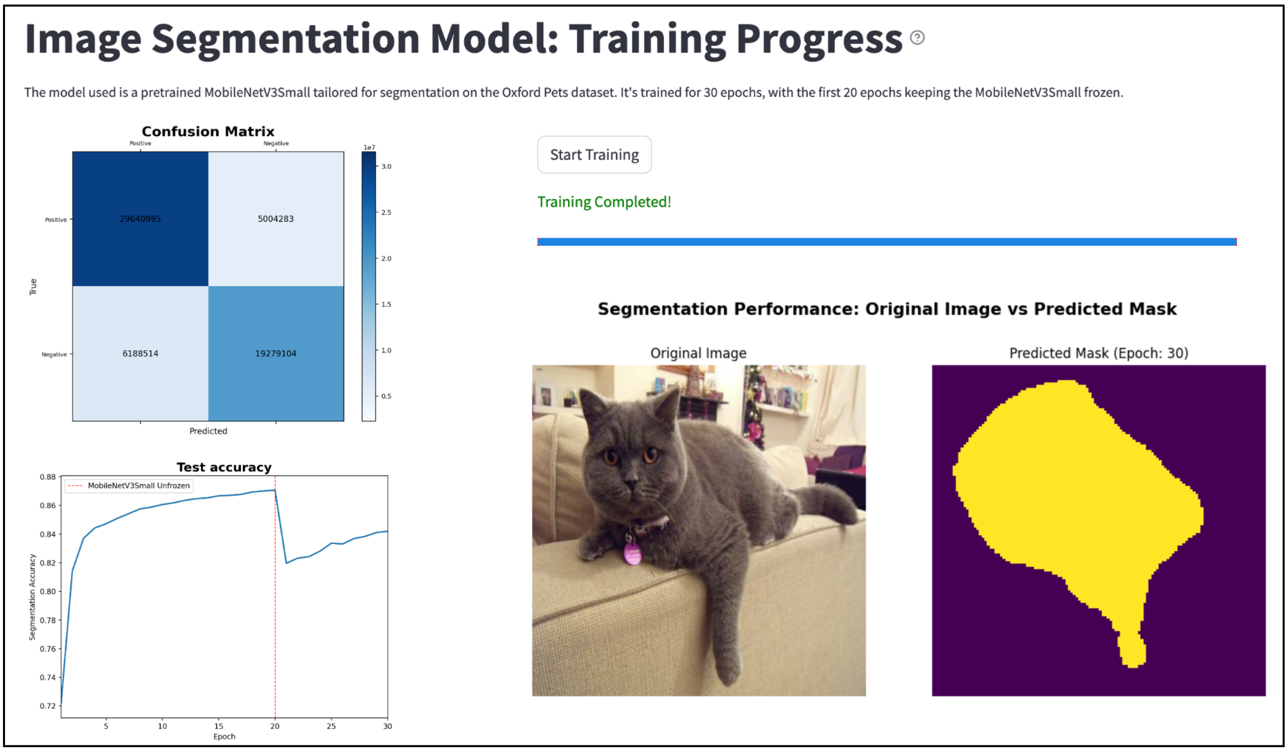


Figure 1. Visualisation output: Web-based dashboard

However, to view the visualisation as discussed in this project, users are advised to adjust the dashboard settings to "**Wide Mode**" and opt for the "**Light Theme**". These recommended settings ensure that the visualisation aligns with the original design and intent. This need for manual adjustment arises due to Streamlit's default configuration constraints.

3.5. Visualisation interpretation & analysis

3.5.1. Confusion Matrix

The use of an interactive confusion matrix for this visualisation is thematically appropriate and highly relevant in the domain of machine learning. The confusion matrix is a staple in classification problems, notably image segmentation, and it succinctly conveys vital information such as True Negatives, False Negatives, True Positives, and False Positives. This matrix effectively showcases the model's performance and its nuances in prediction. Leveraging a well-recognized visualisation technique ensures the conveyed message is easily understood by professionals in the field. Its format is both familiar and informative, streamlining communication

Unlike many traditional visuals, this matrix is not static. With each epoch, it updates in real-time, providing immediate feedback on model training progress. This dynamism is especially crucial for fine-tuning and iterative model development.

In terms of visual elements:

- **Colour Scale**: A dominant dark blue was selected, consistent with the line graph, ensuring uniformity across visualisations. This fixed colour scale across epochs allows users to easily track model evolution without colour distractions.
- **Text**: Clean, sans-serif font ensures maximum readability without any stylistic distractions.
- **Background**: The neutral white background ensures that the colour-coded matrix stands out, further aiding clarity and focus.

All these choices, from the type of visual to the design elements, contribute to the effectiveness of the visualisation that will be discussed further in section **4. "Visualisation Effectiveness".**

3.5.2. Test Accuracy Line Graph

The choice of using a line graph to depict the accuracy over epochs was primarily influenced by the nature of the data. Epochs represent sequential, temporal data points, making a line graph an ideal choice to effectively capture and display trends over time. This ensures that users can easily discern patterns and trajectories in the accuracy of the model. Line graphs offer a clear, unambiguous representation of data. For a metric as vital as model accuracy, it's imperative to have a representation that is straightforward. The graph provides a lucid depiction of how accuracy evolves, offering a comprehensive grasp of model performance as it progresses.

The graph is not merely a static representation; it is designed to be dynamic, updating in real-time after each epoch. This mirrors the actual performance of the model as it trains, enhancing user engagement and offering immediate insights into the performance at each stage. Such an approach makes the experience both informative and interactive for the users.

Furthermore, the inclusion of the red dashed line, indicating the point where "**MobileNetV3Small Unfrozen**", serves as a critical visual marker. It signals a potential shift in model training, hinting at changes in performance and adding an additional layer of contextual information to the graph.

In terms of visual elements:

- **Colour:** A white background was chosen for clarity, with the accuracy data depicted in a contrasting dark blue line, ensuring focus and prominence.
- **Lines:** Beyond the primary accuracy representation, an interrupted red line signifies the point where "MobileNetV3Small" was unfrozen, offering a clear delineation of training phases.
- **Text:** Black sans serif font provides a clean, academic appearance, enhancing readability.

All these design choices aim to maximise the effectiveness of the visualisation,that will be delved deeper into in section **4, "Visualisation Effectiveness"**.

3.5.3. Segmentation Performance

The "Image vs Predicted Mask" visualisation has been selected to transparently display the model's performance on actual images from the testing set. The primary advantage of this visualisation is its ability to reduce cognitive load. By presenting the original image alongside its predicted mask, it makes the model's precision immediately noticeable, catering even to users with limited understanding of the model or the problem.

The visualisation's dynamic nature continues to be a highlight, allowing real-time understanding after each epoch. This dynamic update mechanism provides users with a continuous insight into how the model improves its predictions over time.

In terms of visual elements:

- **Image**: The use of a British shorthair cat is strategic. Its easily recognizable form ensures that users can instantaneously discern the subject, eliminating any undue cognitive effort in image interpretation.
- **Text**: Keeping with the theme of simplicity and directness from the previous visualisations, the textual elements are straightforward, ensuring clarity.
- **Colour**: A notable change in this visualisation is the adoption of the viridis colour palette for the mask. This choice is not arbitrary. Viridis is a preferred selection in image segmentation tasks because of its clarity in demarcating features. While different from the dark blue used in prior visualisations, viridis serves a clear functional purpose, helping users quickly understand the model's output without confusion.
- **Background**: The neutral white background remains consistent, ensuring that all attention is directed towards the primary visual content without any distractions.

3.6. Insights and knowledge

3.6.1. Insights from the visualisations

The visualisations offer audiences an invaluable, comprehensive perspective into the intricacies of our model's performance trajectory throughout its training phases.

In the initial training phases, there is a marked increase in performance from the first to the fifth epoch. This swift rise suggests that the model is efficiently adapting to the fundamental features present in the training data.

Moving past these early stages, the model's performance progression becomes more subdued. After its initial growth, there is a clear stabilisation in its metrics, with only marginal improvements observed after reaching an accuracy benchmark of 84%. This behaviour implies that the model has transitioned from understanding broader data patterns to fine-tuning its predictive capacities to capture more intricate nuances.

An area of particular interest is the model's response to the unfreezing of the MobileNetV3Small pretrained base model. Although there's an initial drop in accuracy, this trend is consistent with established behaviours in transfer learning. As the model adjusts the weights of the pretrained layers, it might momentarily destabilise its predictions. However, as training continues, this flexibility allows the model to optimise its performance further.

Analysing the segmentation visualisations, it becomes evident that relying solely on numerical metrics can be limiting. Despite potential variability in numerical accuracy, the model's adeptness at predicting the cat's form remains steadfast. This underlines the importance of complimenting numerical evaluations with visual assessments.

The observations derived from the visual data are further solidified by the accompanying confusion matrix. This matrix not only reflects the visual data trends but also introduces an empirical dimension, validating the insights obtained.

In sum, these visualisations offer a multifaceted understanding, granting a richer perspective on the model's performance across different training scenarios.

3.6.2. Comparative Advantages

Direct and Intuitive Interpretation: Unlike the fragmented and static visualisations often provided by tools like Keras and Matplotlib, the dynamic nature of the presented visualisations offers an immediate and clear insight into the model's performance trajectory.

Holistic Understanding: The visualisations encompass both the quantitative and qualitative aspects of model performance. This holistic representation ensures that users get a rounded perspective, balancing hard metrics with visual results.

Real-time Insights with Pretrained Basemodel: One of the standout benefits of these visualisations is the ability to track real-time effects when integrating pretrained basemodels. Observing the immediate impact and subsequent recovery provides a deeper understanding of how such base models can aid in model performance.

In sum, the visualisations not only showcase the model's performance in an easily interpretable manner but also emphasise the nuanced aspects that might be overlooked in conventional static representations. They serve as a bridge, connecting raw data to actionable insights, benefiting the proposed target audiences.

4. Visualisation effectiveness

4.1. Visualisation type

Effective visualisations hinge on the strategic selection of appropriate visualisation types. These choices should not only complement the data at hand but also vividly convey the intended message. A guiding framework from Keller & Keller (1994, p. 183) delineates seven distinct categories of visualisation goals, serving as our foundational reference in defining our visualisation objectives.

Line Graph (Testing Accuracy Over Epoch):
The primary goal here was to locate values across a temporal span and reveal evolving trends. Given our data and objectives, the line graph emerged as an optimal choice. It effectively displayed the trajectory of training metrics across epochs, tracing the evolution of testing accuracy. As Knaflic (2015, p. 45) articulates, line graphs are preeminent for plotting temporal data. Their inherent structure not only implies a connection between data points but also demystifies underlying trends. In our context, the line graph not only accommodated the available data but also robustly achieved its visualisation goals.

Confusion matrix:
With the goal of distinguishing different classifications in the image, specifically pixels with the animal and without. Utilising the confusion matrix, a quintessential tool in classification tasks, our aim was to provide an immediate snapshot of model performance. Such a representation ensures the viewer can readily grasp the model's precision.

Performance Visualisation (Original vs Predicted Mask):
Our pursuit here was twofold: to both compare and reveal the disparities between predicted results and the actual image. By juxtaposing the original image with its corresponding predicted mask, we endeavoured to offer a stark, direct comparison of the model's segmentation prowess. This approach resonated profoundly with Keller & Keller's fifth and seventh categories (Keller & Keller, 1994), cementing our stance on visual effectiveness through a juxtaposition of revealing and comparing.

4.2. Visual Elements

4.2.1. Text

Text is one of the most foundational elements of a visualisation, often offering context and aiding comprehension. For our visualisation, which has an academic and scientific tenor, we meticulously made design decisions regarding font, size, and style.

Font Choice: Throughout the visualisation, a consistent sans serif font was employed for all titles, subtitles, and contextual annotations. Research by Bernard (2001) highlighted the clear preference of participants for sans serif fonts like Arial when reading on computer screens, especially for students and young adults - a key segment of our audience. Moreover, sans serif fonts are widely regarded as the preferred choice in scientific visualisations, further reinforcing our decision to opt for this font style.

Font Size: Maintaining a uniform font size across the visualisation was another essential component of our strategy to ensure effectiveness. By avoiding variations, we eliminated potential distractions, ensuring that the audience remains focused on the core data and insights. This uniformity in size amplifies clarity and reinforces the visual hierarchy, directing the viewer's attention seamlessly from one data point to another.

Font colour: In our visualisations, the primary font colour used is black. This decision is anchored in its established simplicity and the findings of Zorko (2017) where it stated that the combination of black text on a white background is the most readable combination of colours. Yet, in instances where emphasis was paramount, notably with epoch numbers, a vibrant red was employed. This not only accentuated the key information but also ensured that the colour, while eye-catching, did not deviate from the primary objective of lucid presentation and remained distraction-free.

Contextual Cues: To bolster the richness of our visualisation, contextual cues were judiciously integrated. Annotations such as "The model used is a pre-trained MobileNetV3Small tailored for segmentation on the Oxford Pets dataset. It's trained for 30 epochs, with the first 20 epochs keeping the MobileNetV3Small frozen" and an apt descriptor "MobileNetV3Small Unfrozen" at epoch 20 furnished viewers with added layers of understanding. Midway (2020) proposed the Principle 8 Simple Visuals, Detailed Captions and emphasised that in many instances, figures cannot stand in isolation and necessitate an accompanying detailed explanation. Hence, our visualisations, rooted in this principle, seek to provide a comprehensive understanding, marrying the visual with the textual.

By underpinning our textual decisions with empirical evidence and best practices, we optimised the effectiveness of our visualisation, ensuring that the audience's attention is continually directed towards understanding and interpreting the data.

4.2.2. Colour

The judicious use of colour in visualisations can be potent. As Midway (2020) articulated, colour inherently conveys information, be it directly or subtly. Within our visualisations, we have adhered to a predominantly white background, a deliberate choice aimed at augmenting readability and ensuring our graphics captivate attention without being overbearing. This simple and clear style is common in scientific visuals because it helps make things clear and easy to understand.

The use of dark blue for the line in our graph was an intentional choice to ensure it contrasts sharply against the white background, enhancing its visibility. Such a colour choice is prevalent in Machine Learning visualisations, aiding in line graph interpretation due to its familiarity within the audience.

Regarding the confusion matrix, our adoption of a dark blue scale served a dual purpose. By utilising a sequential colour scheme, as recommended by Midway (2020), where hues transition from light to dark, viewers are intuitively directed to associate darker shades with increasing data values. Beyond its aesthetic value, the dark blue was leveraged as a strategic instrument to amplify clarity, effectively steering the viewer's comprehension of data trends. Furthermore, we maintained a consistent colour scale across all epochs. This ensures that, irrespective of the maximum value in any given epoch, colours remain constant, allowing for an unambiguous visual representation of the model's evolution from the first to the thirtieth epoch.

4.2.3. Image

In the segmentation performance visualisation, the strategic choice to showcase the original image of the British Shorthair cat alongside its predicted mask serves a dual purpose, both enhancing clarity, effectiveness and reducing cognitive load for the viewer.

Tufte (2001) emphasises that the essence of graphical excellence lies in the ability to present complex ideas with utmost clarity, precision, and efficiency. This excellence is achieved when information is conveyed in the shortest time, utilising the least amount of space and ink. By juxtaposing the original British Shorthair cat image with its corresponding predicted mask, the visualisation perfectly captures Tufte's principles. Viewers can immediately gauge the model's performance, discerning its accuracy and pinpointing areas of discrepancy without the need for additional explanatory text or legends.

Furthermore, the use of the British Shorthair cat's real image does more than offer clarity - it forges an emotional connection. Recognizable and tangible, the cat image naturally engenders viewer engagement. Rather than being mere passive observers, the audience becomes actively involved in processing the information, further aided by the direct comparison available to them. By providing a clear visual reference through the actual subject of the segmentation, the visualisation eliminates the audience's need to mentally reconstruct or imagine the original from the predicted mask alone. This approach, especially pivotal when navigating the intricacies of machine learning model outputs, ensures a streamlined and enriched viewing experience.

4.3. Dashboard Design

The dashboard is meticulously designed to encapsulate a wealth of information, seamlessly merging various visual elements to foster a holistic understanding of the model's progression. By integrating both quantitative and qualitative insights, it transcends traditional data presentations. Instead of isolated numbers or standalone images, the dashboard offers a synchronised view where every piece contributes to a greater understanding.

This approach is heavily influenced by Tufte (2006) where he provided great examples of bringing together textual, visual, and quantitative information for creating effective visualisations. Our dashboard embodies this ethos. Rather than dispersing insights across disparate components, we've amalgamated them into a unified platform. This not only simplifies the user experience but also augments comprehension. By viewing all pertinent information in tandem, users can easily understand the model progression during the training progress.

Utilising a wide frame for the visualisations significantly enhances the effectiveness. By occupying the full width of the webpage, each detail is prominently displayed, ensuring no data nuances are missed. Moreover, by fitting the visualisations snugly to the screen's dimensions, we channel the user's attention directly onto the data. This immersive experience minimises external distractions, ensuring that users remain focused solely on the visualisations. The wide frame thus becomes more than a mere design choice; it's a strategic decision aimed at optimising user engagement and comprehension. Every inch of screen real estate is meticulously employed to tell the data's story, making the user's journey both immersive and enlightening.

A primary design principle guiding the development of this dashboard is simplicity. Given the inherently scientific nature of the visualisations and the objective of fostering learning, it was imperative to adopt an approach that prioritised clarity. As Tufte (2001) sagely points out, the real challenge in information design lies not in adding layers of complexity to the simple but in portraying the inherently complex in an accessible and clear manner. This design ethos echoes Tufte's sentiment about the role of design in "the revelation of the complex", ensuring that even those unfamiliar with the nuances of machine learning can derive meaningful insights from the dashboard. Through its simple yet effective design, the dashboard achieves the lofty goal of turning complexity into clarity, embodying the very essence of effective scientific visualisation.

The dashboard's interactivity significantly elevates its utility and effectiveness. By introducing the "Start Training" button, users gain agency over the visual experience, choosing precisely when to initiate the training simulation. Coupled with real-time dynamic updates, this feature offers a vivid and accurate representation of the model's training journey. Weissgerber (2016) emphasised that interactive graphics provide insights that static visuals cannot convey, proving invaluable in promoting transparency and reproducibility. In line with this, our dashboard's interactive dimension not only captivates users but also unravels the complexities inherent in the training phase, achieving the project's overarching objective of clarity and comprehension.

5. Project Tasks and Timeline

Task Name	Duration	Start	Finish
Literature Review and Dataset Familiarisation	10 days	Tue 25/07/23	Mon 7/08/23
Oxford Pets Dataset Preprocessing and Augmentation	7 days	Mon 7/08/23	Tue 15/08/23
Model Training and Fine-tuning	7 days	Wed 16/08/23	Thu 24/08/23
Development of Visualisation Prototypes	6 days	Fri 25/08/23	Fri 1/09/23
Integration of Model Outputs with Visualisation Tools	7 days	Mon 4/09/23	Tue 12/09/23
Dashboard Refinement and Feature Addition	14 days	Wed 13/09/23	Sat 30/09/23
Testing and Debugging	7 days	Sat 30/09/23	Sat 7/10/23
Final Presentation	2 days	Fri 6/10/23	Mon 9/10/23
Final Report & Documentation	8 days	Wed 11/10/23	Fri 20/10/23

Table 1. Project Tasks Table

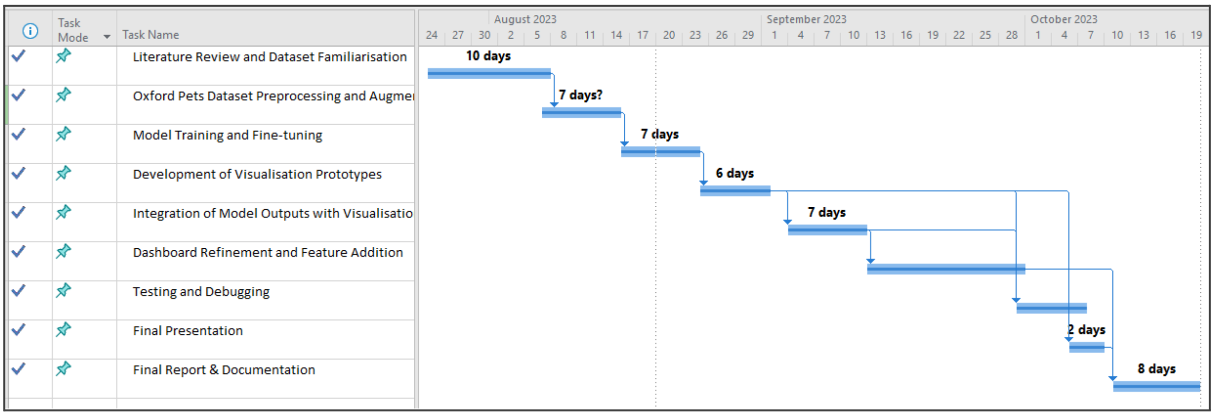


Figure 2. Project Tasks Gantt Chart

6. Reflections

6.1. Achievements

Throughout the course of this project, we met several goals that were initially set out:

Development of a Web-Based Visualisation: We successfully developed a web-based visualisation that is now easily accessible to anyone, especially catering to our target audience - students in data science, IT, and machine learning fields. This visualisation acts as a great example of model training progress for their learning.

Introducing Dynamic Features: By incorporating dynamic visualisations, we directly addressed and overcame the shortfalls of existing tools in the market. This real-time, dynamic feature significantly bolsters users' comprehension and engagement with the data.

Enhancing User Experience: We enhanced the visualisation with interactive elements, permitting users to actively engage with the visualisations. This not only enriches their user experience but also amplifies their learning curve.

Timely Project Completion: Despite various challenges, we completed the project within the allocated time for the assessment, showcasing our commitment and effective planning.

6.2. Challenges & Shortcomings

Dynamic Visualisation with Matplotlib: One of our main challenges was integrating dynamic visualisation capabilities using Matplotlib, a library primarily tailored for static visualisations. The process of converting static plots into dynamic ones was arduous and time-consuming, demanding meticulous planning and execution.

Acclimatisation with Streamlit: As we explored new tools, we chose to employ Streamlit, a platform we hadn't worked with before. The process of understanding its features and optimising its capabilities led to an extension in our initial projected timeline.

Our initial project plan estimated the development of the dashboard using Streamlit to span 7 days. However, due to unforeseen complexities and the steep learning curve associated with

Streamlit, the actual time taken doubled to 14 days. This adjustment was necessary to ensure the quality and functionality of the final product.

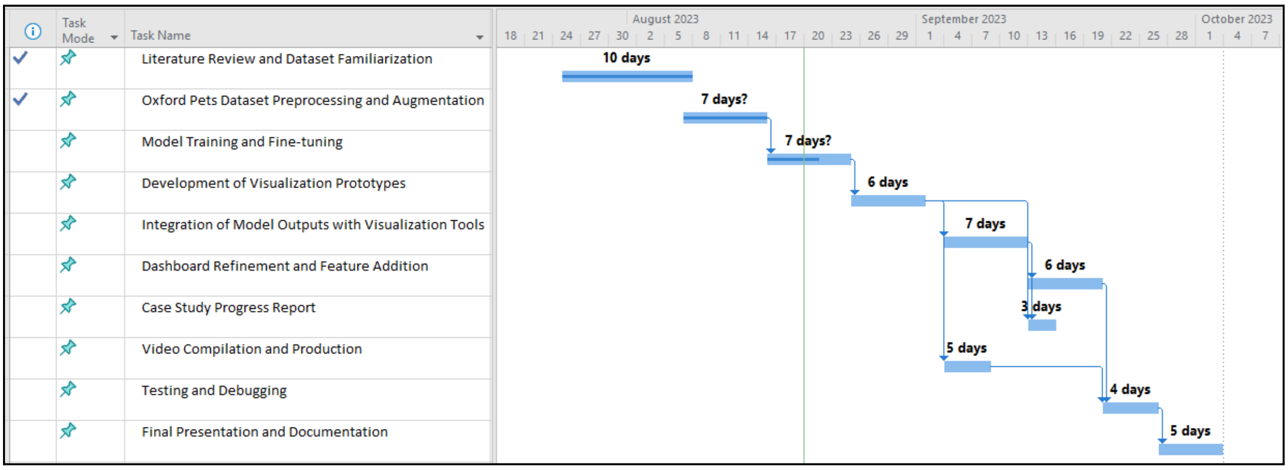


Figure 3. Tasks Gantt Chart in the proposal

Initially, our intention was to incorporate tooltips for every visualisation to provide comprehensive insights and enhance user understanding. Unfortunately, due to limitations in Streamlight's functionality, we were unable to implement this feature, which could have contributed to a more effective user interaction with the visualisations.

Omission of Video Component: Initially, our project proposal included the development of a video component. However, as the project evolved, we realised that a website would be more suited to our visualisation objectives. This pivot was made in the best interests of the project's goals and the target audience's needs.

6.3. Lessons Learned

Throughout the duration of this project, several lessons were learned that have invaluable implications for future endeavours:

Deepened Understanding of Effective Visualisation: Through this project, I gained a richer appreciation for the subtleties of effective visual representation. Elements such as text selection, the choice of visualisation types, colour schemes, and dashboard layout play pivotal roles in conveying information accurately and engagingly. This knowledge will undoubtedly influence and enhance my future data analysis and scientific visualisation projects.

Enhancing Technical Proficiency: Engaging with Streamlit and Matplotlib deepened my understanding and competency in the realm of data visualisation. This experience enriched my skill set, positioning me more favourably for a future career in data analytics or science.

Navigating Project Management: Managing an ambitious project within a tight timeline emphasised the criticality of effective time and task management. This project served as a practical lesson in balancing ambitious goals with realistic timelines, and the importance of flexibility in the face of unforeseen challenges.

Consolidation of Machine Learning Knowledge: Through the visual representation of machine learning models, I garnered a more profound understanding of their training processes. This not only consolidated my existing knowledge but also offered a deeper perspective on the intricacies of model training,

7. Project Journal

Date: 28/07/23: Activity: Conducted a comprehensive literature review focusing on segmentation techniques and datasets.
Approximate Time Taken: 5 hours

Date: 10/08/23: Activity: Began preprocessing the Oxford Pets dataset; standardised image sizes and corrected anomalies.
Approximate Time Taken: 6 hours

Date: 12/08/23: Activity: Conducted data augmentation to increase the dataset diversity, implementing random flips and rotations.
Approximate Time Taken: 4 hours

Date: 18/08/23: Activity: Initialized the training of the model using MobileNetV3Small as a base.
Approximate Time Taken: 7 hours

Date: 20/08/23: Activity: Optimised hyperparameters for the model and evaluated initial training results.
Approximate Time Taken: 5.5 hours

Date: 27/08/23: Activity: Drafted the initial design for visualisation prototypes on Canva; focused on dashboard layout and interactivity.
Approximate Time Taken: 6 hours

Date: 06/09/23: Activity: Integrated the trained model's outputs with the visualisation.
Approximate Time Taken: 6.5 hours

Date: 18/09/23: Activity: Created the draft dashboard.
Approximate Time Taken: 6 hours

Date: 25/09/23: Activity: Refined Dashboard and Added features (Button/Progress Bar)
Approximate Time Taken: 8 hours

Date: 27/09/23: Activity: Final Testing
Approximate Time Taken: 2 hours

Date: 30/09/23: Activity: Set up github repository
Approximate Time Taken: 2 hours

Date: 7/10/23: Activity: Presentation Preparation
Approximate Time Taken: 5 hours

Date: 15/10/23: Activity: Wrote final report
Approximate Time Taken: 5 hours

Date: 25/10/23: Activity: Finalised final report

Approximate Time Taken: 2 hours

8. Conclusions

Throughout the course of this project, our primary goal of "Bringing clarity to model training" was successfully achieved. The once ambiguous training phase was transformed into a comprehensible visual journey, shining light on the intricate process of model training. This transformation was facilitated through the adept use of a diverse toolset, specifically tailored to foster effective visualisations. Drawing from established visualisation theories and principles, we constructed visualisations that were not only effective but built from basic foundational elements.

Our approach did not stop at merely creating visualisations; we successfully implemented real time model training simulation. This real-time feedback mechanism allowed for a more dynamic understanding of the model's training process. To ensure a seamless user experience, we developed a dynamic web-based visualisation platform, harnessing the best practices of modern web development and visualisation techniques.

In hindsight, one area of potential improvement would have been allocating more time to understand Streamlit during the earlier stages of the project. The focus in August was predominantly on dataset work. A more proactive exploration of Streamlit might have led to the incorporation of additional features or tools, enriching the overall project.

For future endeavours, the project has the potential to expand its visualisation scope to other areas of machine learning, such as Classification and object detection. Introducing various model architectures, like VGG-net, could offer a broader perspective for learners. Additionally, enhancing user interactivity by allowing dynamic parameter adjustments, such as changing epochs or learning rates, would offer users a hands-on experience, observing firsthand how different parameters influence the model's evolution.

9. References

1. Keller, P. R., Keller, M. M., Markel, S., Mallinckrodt, A. J., & McKay, S. (1994). Visual Cues: Practical Data Visualization. *Computers in Physics*, *8*(3), 297–298. https://doi.org/10.1063/1.4823299

2. Knaflic, C. N. (2015). *Storytelling with data : a data visualization guide for business professionals* (1st edition). Wiley.

3. Midway, S. R. (2020). Principles of Effective Data Visualization. *Patterns*, 1(9), 100141. https://doi.org/10.1016/j.patter.2020.100141

4. Kaspar, K., Wehlitz, T., von Knobelsdorff, S., Wulf, T., & von Saldern, M. A. (2015). A matter of font type: The effect of serifs on the evaluation of scientific abstracts. *International journal of psychology : Journal international de psychologie*, *50*(5), 372–378. https://doi.org/10.1002/ijop.12160

5. Tufte, E. R. (2001). *The visual display of quantitative information* (2nd ed.). Graphics Press.

6. Bernard, M., Liao, C. H., & Mills, M. (2001). The effects of font type and size on the legibility and reading time of online text by older adults. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems* (pp. 175-176). Association for Computing Machinery. https://doi.org/10.1145/634067.634173

7. Tufte, E. R. (2006). *Beautiful evidence*. Graphics Press.

8. Anja Zorko, Snježana Ivančić Valenko, Damira Keček, Darijo Čerepinko, & Mario Tomiša. (2017). The impact of the text and background color on the screen reading experience. *Tehnički Glasnik*, *11*(3), 78–82.

9. Weissgerber, T. L., Garovic, V. D., Savic, M., Winham, S. J., & Milic, N. M. (2016). From Static to Interactive: Transforming Data Visualization to Improve Transparency. *PLOS Biology, 14*(6), e1002484. https://doi.org/10.1371/journal.pbio.1002484

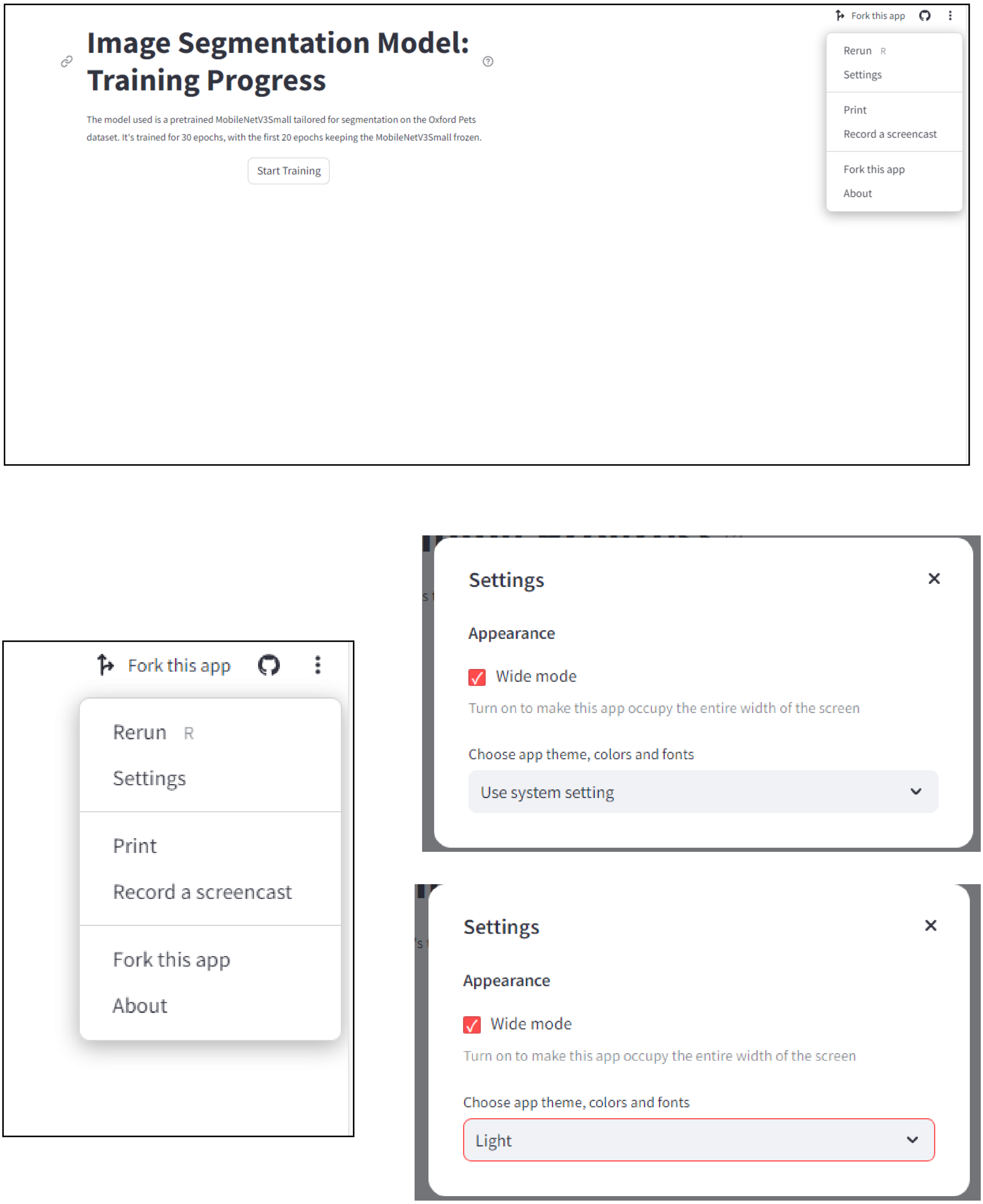
10. Appendices

10.1. Appendix 1: Visualisation Settings Guide

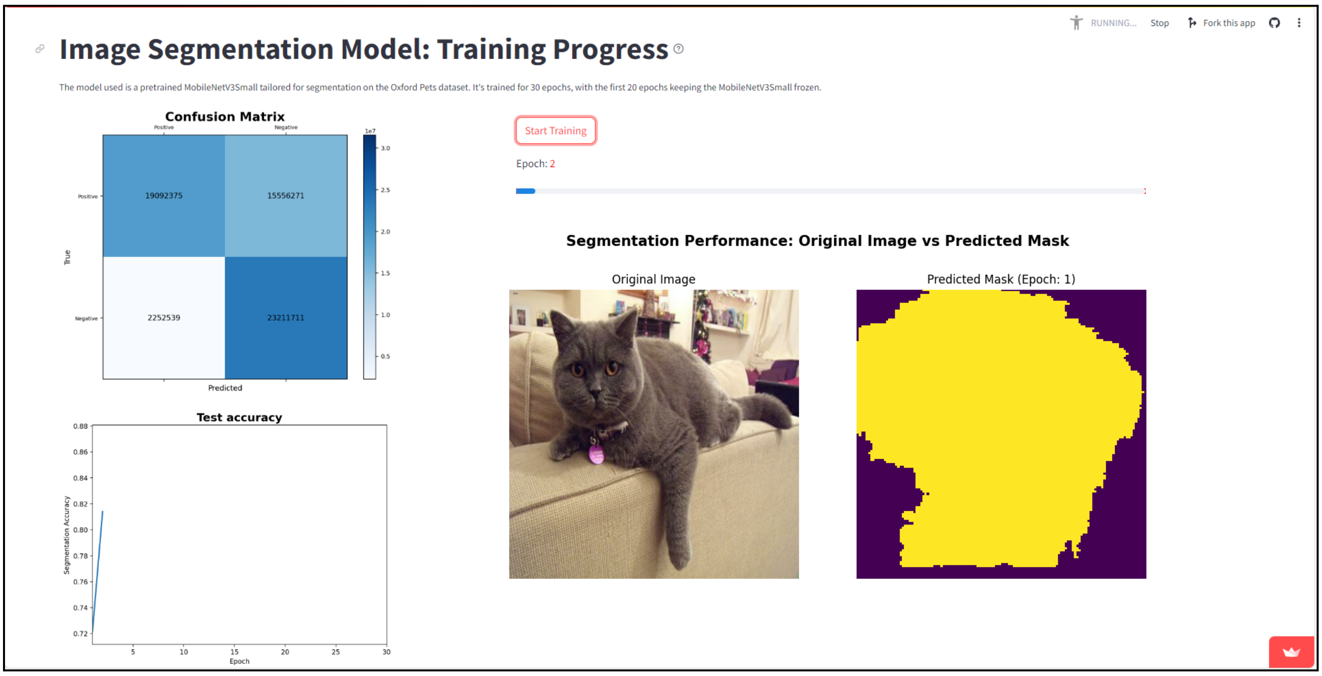
To ensure that you have the best user experience and optimal viewing of our visualisation, please follow the guidance below to adjust the settings:

Wide Mode Activation:
- Navigate to the Settings icon, typically located at the top right corner of the visualisation dashboard.
- Under the Appearance section, you will find an option labelled Wide mode.
- Activate the Wide mode by toggling the switch. This allows the visualisation to span across the entire width of your screen, ensuring all details are clearly visible without any need for horizontal scrolling.

Choosing the Theme:
- Still under the Appearance section, locate the dropdown menu where you can choose the app theme.
- From the available options, select Light. This will change the theme to a lighter colour scheme, which enhances readability and reduces eye strain during extended viewing.



Figs. Wide Mode Activation & Light Theme Selection



Figs. The dashboard with the correct settings