University of Southampton

Faculty of Engineering and Physical Sciences

Electronics and Computer Science

Classification of Mushroom Images using Fine-grained Image Processing Methods

by

Alexander R Dennington - ad2n18@soton.ac.uk - 29534267

Supervisor: Dr Manuel Leon Urrutia - University of Southampton
Second Examiner: Professor Richard A Watson

A dissertation submitted in partial fulfilment of the degree

of MSc Data Science

Statement of Originality

- I have read and understood the ECS Academic Integrity information and the University's Academic Integrity Guidance for Students.
- I am aware that failure to act in accordance with the Regulations Governing Academic Integrity may lead to the imposition of penalties which, for the most serious cases, may include termination of programme.
- I consent to the University copying and distributing any or all of my work in any form and using third parties (who may be based outside the EU/EEA) to verify whether my work contains plagiarised material, and for quality assurance purposes.

You must change the statements in the boxes if you do not agree with them.

We expect you to acknowledge all sources of information (e.g. ideas, algorithms, data) using citations. You must also put quotation marks around any sections of text that you have copied without paraphrasing. If any figures or tables have been taken or modified from another source, you must explain this in the caption and cite the original source.

I have acknowledged all sources, and identified any content taken from elsewhere.

If you have used any code (e.g. open-source code), reference designs, or similar resources that have been produced by anyone else, you must list them in the box below. In the report, you must explain what was used and how it relates to the work you have done.

I have not used any resources produced by anyone else.

You can consult with module teaching staff/demonstrators, but you should not show anyone else your work (this includes uploading your work to publicly-accessible repositories e.g. Github, unless expressly permitted by the module leader), or help them to do theirs. For individual assignments, we expect you to work on your own. For group assignments, we expect that you work only with your allocated group. You must get permission in writing from the module teaching staff before you seek outside assistance, e.g. a proofreading service, and declare it here.

I did all the work myself, or with my allocated group, and have not helped anyone else.

We expect that you have not fabricated, modified or distorted any data, evidence, references, experimental results, or other material used or presented in the report. You must clearly describe your experiments and how the results were obtained, and include all data, source code and/or designs (either in the report, or submitted as a separate file) so that your results could be reproduced.

The material in the report is genuine, and I have included all my data/code/designs.

We expect that you have not previously submitted any part of this work for another assessment. You must get permission in writing from the module teaching staff before re-using any of your previously submitted work for this assessment.

I have not submitted any part of this work for another assessment.

If your work involved research/studies (including surveys) on human participants, their cells or data, or on animals, you must have been granted ethical approval before the work was carried out, and any experiments must have followed these requirements. You must give details of this in the report, and list the ethical approval reference number(s) in the box below.

My work did not involve human participants, their cells or data, or animals.

Abstract

Mushrooms are notoriously difficult to identify, and incorrect classification can result in serious illness or death. Additionally, the risks of misidentification have led to a disconnect between many western cultures and this useful tool. Therefore, the development of highly accurate methods of mushroom species identification is an important problem to be overcome. In the past, identification relied on visual observation and impractical microscopic analysis. However, with the rise of deep learning, it has now become possible to train computer algorithms to learn to distinguish between the species. In this paper, a novel form of computer vision algorithm called a Vision Transformer is applied to the realm of mushroom image classification. Utilising the MO106 image-set, an accuracy of 94.9% is achieved with the SWin Vision Transformer, surpassing the previous benchmark of 92.6% on the same image-set. This shows new innovations in computer vision can reliably improve upon existing benchmarks in the realm of mushroom classification.

Contents

1 Introduction

1.1 Background

The task of classifying mushrooms has been left to the collective wisdom of humanity for millennia. Be it for food, medicine, poison, clothing or any other of their abundant applications, being able to identify mushrooms has always been an important aspect of human existence. Over time, we have learnt to differentiate between them based on appearance, spore-prints, chemical tests and microscopic features, as well as geographical information such as location, local habitat and seasonal patterns. These different methods will be explored in more detail in section 2.4.

The rise of computers and the internet, combined with photographic technologies, created a new way for mycologists to communicate and share discoveries by uploading images to online forums and community websites, thereby helping each other to identify species and pass on their hard-earned knowledge. More recently, image-processing and classification technologies have been developed, a watershed moment being the landmark paper "Convolutional networks for images, speech, and time series (1995)" [1] by Bengio and LeCun which introduced the wider computational science community to convolutional neural networks, and led to their adoption as the state of the art for most image processing and image recognition tasks. In more recent years, pre-trained convolutional neural networks have been created - where the model is trained on large ($> 1,000,000$) labeled image-sets of various objects - and which can be utilised for specific tasks via fine-tuning. These pre-trained and fine-tuned models have been found to improve image-task related performances over the basic CNN models, and have been utilised for mushroom image classification tasks. The current state-of-the-art for image classification lies with pre-trained image classifiers such as ResNet[2] and EfficientNet (EffNet)[3]. However, given the high turnover of papers and research into computer vision, there lies an opportunity to further improve the current state-of-the-art using these new methods. In particular, vision transformers [4] are a product of the latest research, having an architecture similar to NLP transformers and relying on similar methodology when applied to images. These are shown to perform well, particularly on fine-grained-image classification tasks, and seem suitable for the task of mushroom image classification. Therefore this research project will be of use to the computer science community as an attempt to set a benchmark in fine-grained image classification performance and demonstrate the current and future utility of Vision Transformers in contributing to this area, and other computer vision-related tasks.

Furthermore, the problem of mushroom identification has never been more relevant, since the topic of mushrooms has boomed in popularity in recent years. This has been fuelled by a drive for more healthy lifestyles and environmentally-friendly methods of food production[17], as well as the need to develop alternative technologies for treating environmental problems (such as the breakdown of plastics/oil-spills) [18], and the serious demand for effective mental health treatments[19], all of which mushrooms contribute under the

umbrella of the "Radical Mycology" movement. Therefore, it would benefit the scientific community to have more accurate and useful tools for identifying mushrooms, since this would help the general public to engage with the science of mycology and feel the warmth of its abundant benefits.

1.2 Vision Transformers

This project makes use of a novel form of computer vision technology called a Vision Transformer. The original vision transformer was created by the authors Dosovitskiy et al. in their paper *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (2021)*[4] in an attempt to transfer the application of transformers from the natural language domain, where they act as the backbone of most language models, to the vision domain. This presented a number of difficulties, due to the significant differences between language and images. There were also scalability issues to overcome related to the use of self-attention mechanisms utilised by transformers, and how attention can be applied to high resolution images compared to much smaller sentences handled in NLP tasks. The result of the above paper was a transformer architecture that, once properly pre-trained, can learn salient features within images in order to classify them. However, the tasks that the ViT could handle were limited to image classification and joint vision-language modeling. Also, ViTs have a fixed patch-size meaning they do not scale up to higher resolutions, often required for object detection and image segmentation. In response to these limitations, the authors Liu et al. developed a differing transformer in their paper *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows (2021).* This vision transformer, commonly referred to as SWin (Shifted Windows), modified and built upon the ViT using a shifted-window self-attention technique, coupled with the merging of patches over subsequent layers in order to learn image representations at different scales. The SWin model was capable of a larger variety of image tasks than the ViT, and offers the ability to fine-tune the transformer on higher resolution images - where the ViT could only work with images of same size as pre-training images). However, what is common with both types of transformer is the need to be pre-trained on a very large data-set of images, otherwise the models fail to generalise well in fine-tuning. Therefore, these models are most commonly used after pre-training on the ImageNet data-set, where the transformers can learn a very general and applicable set of features, which can then be utilised in the fine-tuning phase, very similar to how the state-of-the-art CNNs are used. The workings of these transformers and the underlying algorithms will be discussed in more detail in the Literature Review section of this paper.

1.3 Research Aim

This project intends to build upon existing research in the area of mushroom species image classification by utilising state-of-the-art methods that will improve upon the currently

existing benchmarks. Being able to identify mushroom species to a higher degree of confidence will be of benefit to the general public by enabling them to harness the many valuable medicinal and general-health related properties that they contain, while also providing more reliable assistance in differentiating edible mushrooms from their poisonous counterparts, potentially preventing unnecessary deaths from misidentification.
While image classification performances have been steadily improving year after year, the ability to classify mushroom species is a fine-grained image classification task and remains a clear challenge and useful benchmark for image classifiers. This is due to the homogeneous appearance of many different mushroom species and the differing abundance and locations of each species. The former is a challenge because the image classifier relies on feature extraction based on visual characteristics, which can be overcome by having a large representation of each species in the data-set, which the latter problem then aggravates. For these reasons, many state-of-the-art image-net classifiers perform subpar on mushroom-related tasks, and particular emphasis has to be put on crafting such classifiers into well-performing mushroom classifiers using transfer learning and feature extraction. This paper aims to demonstrate the performance of the current state-of-the-art technology in fine-grained image classification, and participate in a possible renaissance within computer vision technologies. Vision transformers have not yet been applied on such a large data-set of mushrooms and therefore this project seeks to make an original contribution to this area of computer vision research.

1.4 Historical Techniques for Mushroom Species Classification

Over time, humans have learnt various discernible characteristics with which to identify mushrooms. The diagram in Figure 1 shows the different parts of a standard mushroom. The cap can differ in dimension, colour, shape and texture, while the margin of the cap can be one of smooth, irregular or split. The gills under the cap contain the spores, and can differ by attachment (adnate, adnexed, decurrent, distant, emarginate, remote or free), spacing (crowded, close, distant), colour and margin (smooth or irregular). The stem is the vertical support structure of the mushroom and can differ by dimension, shape (equal, tapered downwards/upwards, club-shaped, bulbous), attachment (central, excentric, lateral) and inside (solid, hollow, stuffed). For more specialised and unique species of mushroom (e.g. polypores), the methods for identifying based on appearance will be different. Other methods of identification include taking spore prints (a parallel to how humans take fingerprints, different species of mushroom leave behind a unique spore print), chemical tests, and microscopic features.

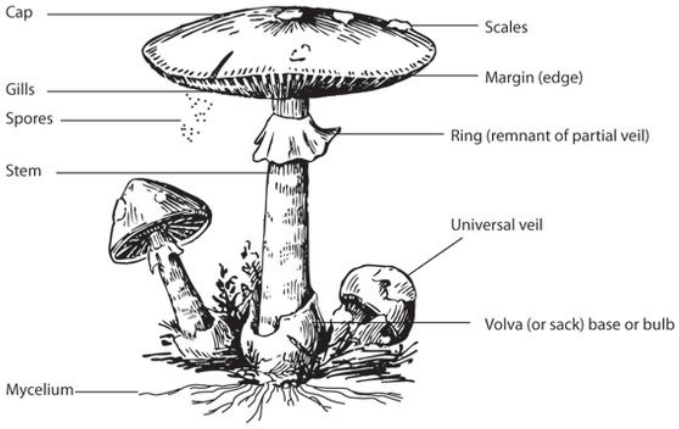


Figure 1: Diagram of Mushroom [sourced from "https://www.mushroomdiary.co.uk/mushroom-identification/"]

2 Literature Review

2.1 Introduction

Mushrooms are an often overlooked and feared part of the natural world. Their abundant benefits to many different areas relevant to human existence will not be harnessed unless the relationship between humanity and fungi is to improve. For a good basic overview of this subject, see *Mycelium Running: How Mushrooms Can Help Save the World (2005)* by Paul Stamets. Different cultures have evolved a different relationship with fungi starting from pre-civilisation. It has been observed that people in Western cultures (with the exception of South America) are more likely to fear mushrooms for their poisonous attributes, and haven't widely adopted them into their routine existence. In contrast, those in Eastern cultures, such as people in Russia and China, have been using mushrooms as part of their daily routine for millennia, as part of cooking, medicine, farming and religious rituals. In collecting these useful tools, these cultures developed their own rules of identification, which are of critical importance, since a misidentified mushroom can lead to serious illness and often death. However, the advent of machine learning has enabled an alternative method for identification, that allows people with very little mycological knowledge to be able to identify the useful mushrooms among the many poisonous or idle varieties. While these methods will not come close to competing with the ingrained knowledge of thousands of years of mushroom use contained within an expert practitioner, they will provide accessibility to many previously fearful but interested amateurs.
In this project, a reliable data-set containing high quality images of mushrooms belonging to one of 106 different species will be combined with the latest image classification techniques to build a more accurate model for classifying these mushrooms. The following sections

will detail the latest literature in mushroom classification as well as the relevant literature on image classification, with an emphasis on fine-grained image classification.

2.2 Mushroom Classification

Mushrooms have been the focus of many research papers in the past, attempting to develop and test the performance of algorithms that are trained to work with fine-grained-image sets such as mushrooms. While there are papers that are more general in their approach to dealing with mushroom images, using more traditional machine learning methods, these appear to lack the performance potential of deep learning techniques such as CNNs and Vision Transformers. At the time of writing and after an in-depth search, it has been found that only one previous research attempt has been made to create a mushroom classifier utilising vision transformers, which is covered in this review.
The winning entry at the FGVCx (Fine-grained Visual Categorization) Fungi Classification Challenge at the CVPR (Computer Vision & Pattern Recognition) Conference 2018 [5] was an ensemble model consisting of 6 pre-trained CNNs. Three types of CNN were used (Inception-v4, Inception-v4 "x2" and Inception-ResNet-v2) and fine-tuned on two large image sets (ImageNet2012 and LifeCLEF2018). This ensemble was then trained on the FGVCx fungi classification challenge data-set covering 1394 species and made up of 85578 training images, 4182 validation images, and 9758 privately-labelled test images. Other techniques used by the authors to improve model performance include test-time augmentation and adjusting the prior-distribution of the training set prior to training. The best model produced by the authors had a validation accuracy of 60.3% on the validation set of mushroom images. This is an impressive performance given the model is trying to classify the images into 1394 species, a lot of which are hardly discernible to the human eye. However, this accuracy is not good for practical purposes since a wrong identification could prove very costly. A way to improve this accuracy would be to cut down on the number of species to classify, since a lot of these species will be very uncommon and therefore will be unlikely to appear in nature, as well as being under-represented in the data-set, meaning the model will have enormous trouble learning to classify it. However, due to the nature of the competition it was not possible to clean the data-set as such. Additionally, another way to improve the performance of this model would be to gather more images, especially for the under-represented species, since this would provide more examples to learn features, and a data-set of 85578 images would only provide (on average) 61 images per species.

The authors J. Preechasuk et al. in their paper "Image Analysis of Mushroom Types Classification by Convolution Neural Networks" (2019) [6] decided to train a simple CNN binary classifier on a data-set of 8556 mushroom images, containing 45 species (10 Poisonous, 35 Edible). 6000 of the images contained the *edible* mushrooms while the other 2556 contained the *poisonous* mushrooms. The authors decided to resize the images to (96x96) and then perform various augmentations on them (rotation, zoom, width-shifting,

height-shifting, brightness adjustments and flipping) in order to improve the generalisation performance of their model. The resulting classifier had an impressive F1-score of 0.74. However, this task performs a binary image classification, so therefore is learning the mushroom features only so far as they help discern poisonous from non-poisonous, and not the relevant features to discern the species. This is quite a difficult task since it is not widely thought that visual appearance is significantly correlated with being poisonous. Although most poisonous varieties of mushroom belong to the Amanita family due to the presence of the amanitin toxin, the Amanita family are not well represented within this trial so the classifier won't rely on distinguishing this variety from others. Another area of improvement in this study is the small data-set size, since it uses 8556 images which is small in the age of big data, and a larger number of images can only improve the model. The authors also were not explicitly clear on the size of the validation set, as it is only mentioned once and without an actual number. This reduces the reliability and validity of this paper since they haven't been totally transparent and exhaustive in describing their methodology and modelling process. The low number of species covered by the classifier (45) means it is an impractical model to use. It was initially assumed the authors may have intended on creating a model to classify local varieties within Thailand (the origins of the paper), however on further analysis of the species included in the study, this cannot be true. The proportion of poisonous to edible species is unbalanced, with over 70% of the images being of edible mushrooms and the other 30% being poisonous. Creating a truly unbiased classifier would require a large, well balanced data-set. Lastly, the authors reduced the resolution of all the images to 96x96, which would indelibly cause useful information to be lost from the images. This would have been a decision to increase the training time of the model, at the expense of model performance. However, this study is useful in that it demonstrates good performance in-spite of the compromises that were made, showing the potential of image recognition algorithms to recognise useful features in visibly similar images.

The authors M. Ottom et al. in their paper "Classification of Mushroom Fungi Using Machine Learning Techniques" [7], followed a similar path to the previous authors and decided to classify by whether the mushroom in the image is *poisonous* or *non-poisonous*. However, they relied on a series of more traditional machine learning algorithms such as Neural Networks, SVMs, Decision Trees and K-nearest neighbours to be their classifier, and extracted eigen-features, histogram features and parametric features from the images to be fed into the algorithms. These eigen-features included physical characteristics of the mushroom in the image such as height and width of the mushroom. Using these extracted dimensionality features, the most successful model was the K-Nearest Neighbours model which managed an accuracy of 86%. The authors used a lot of feature engineering to extract the most appropriate features for the models. This included attempting to extract physical characteristics of the mushroom from the image, however the best performing model was the one which relied on the actual physical characteristics rather than their

estimates. Using physical characteristics in the model would be impractical unless working in the field. Also, the authors fail to mention the size of the training and evaluation sets other than to comment on their small size. As commented earlier, it would improve this work to have a larger data-set of images to learn features from. While the task that M. Ottom et al. set out to perform is not identical to the task in this project, their paper does highlight how machine learning algorithms have been applied to mushrooms in the past and the amount of effort required to handcraft features to feed into these algorithms, while using an approach like Vision Transformers can extract features implicitly.

In a similar fashion to this dissertation, the authors Z. Huang et al in "A Multi-Stage Vision Transformer for Fine-grained Image Classification" (2021)[8] decided to experiment with vision transformers in the realm of mushroom classification, as a representation of fine-grained image classification. The authors started with a standard ViT, however, they were dissatisfied with how the structure of the ViT harms the discriminative regions. Therefore they proposed a pooling-based vision transformer with overlapping patches in a hierarchical structure, very similar to how the SWin[13] model works. They then proceeded to fit their modified ViT to a self-curated data-set of mushroom images obtained from field collections and the internet, covering a total of 52 species. The results of their project showed an improvement over the original ViT structure while using fewer parameters and computational units. While the results of this project are promising, the lack of species coverage means the model isn't a useful tool in the field of mushroom classification. Additionally, the modifications they made to the ViT had already been published as the SWin model before their paper was released, so utilising the work in [13] could have been more time-efficient. However, this still provides a useful result in a growing trend of research towards vision transformers for vision-related tasks, and acts as a validation of the SWin model since it was presumably produced independently and arrived at the same technique.

The most relevant paper is "Mushroom Image Classification with CNNs: A Case-Study of Different Learning Strategies" (2021) by N. Kiss and L. Czuni [9] in which the authors curated their data-set from the FGVCx Fungi Classification Challenge 2018 data-set and the Mushroom Observer website database, filtering the images to meet the standards required for their model, and ending up with the MO106 data-set. This contains 29,100 images of 106 different species and is the same data-set to be used in this paper. The authors then proceeded to experiment by fitting different classifier models, whilst utilising the EffNet [3] pre-trained CNN as a backbone (the EffNet is covered later in the review). In total, the authors created 7 different vision models, with various initialisation and training techniques. Firstly a standard EffNet with random weight initialisation was trained from scratch on the data. The images had random augmentations performed on them, similar to this paper, and a stochastic gradient descent optimiser with learning rate 0.01 and sparse categorical cross entropy loss were used. This had a validation accuracy of 48%, and was a useful baseline for the authors. Next, an EffNet-B0 configuration, pre-trained on ImageNet,

was fine-tuned on the data. The best model configuration the authors could produce had a final validation accuracy of 78%. The authors decided at this point to continue down the path of transfer learning, with the next attempt being an EffNet-B5 (larger) variant, using ImageNet weights, and fine-tuned on the data-set. While this model was slower to train, it produced a final validation accuracy of 89.04%. The next step the authors took was to use an EffNet pre-trained on the ImageNet data-set using the Noisy Student training technique [11]. After fine-tuning, the accuracy improved to 90.9%. Going forward, the authors decided to use these weights in their future models, with the next one being a class specific branch model. By using the confusion matrix, the authors could see which pairs of species were most commonly being mistaken, and divide the data into subsets based on this information. They then implemented classifiers on each of these subsets. The accuracy of this model improved to 90.96%, not a big jump. The best performing model was trained using gradual problem growth. This is where the number of classes is gradually increased, reusing the same weights.. The authors started with 6 species, then 15, 26, 55 and 106. The final validation accuracy was 92.6%, and represents the state-of-the-art for mushroom image classification accuracy on the MO106 data-set. This paper is important to the project as it is the explicit aim of this project to better the benchmark set by the authors of 92.6%. The exact steps of the authors to produce the MO106 data-set are detailed in the methodology section, in which they reduced a large set of potential images to 29,100 based on abundance and image quality. The painstaking and advanced process these authors took in producing their data-set is the reason it was chosen for this study, as well as the availability of an obvious benchmark to measure this project's success. The particular characteristics of the final MO106 data-set that are impressive include the high average number of images per species of 275, with a range of 581 to 105, and the resolution of the images not being standardised, since that allows more flexibility in this project to define such parameters.

2.3 Image Classification Algorithms

It is useful to know how the algorithms used in this project, and their relatives, work by analysing the important literature published related to them. The models produced in this paper use certain types of computer vision algorithms called Vision Transformers (also known as ViTs). These are currently considered to be the state of the art in vision related tasks and rely on a methodological framework derived from Natural Language Processing transformers. The evolution of the Vision Transformer started with the ViT, and has been built upon by other works which improved certain elements of the ViT or provided competition through a different architecture. The following sections review these different algorithms.

2.3.1 Vision Transformer (ViT)

The authors of the paper "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" (2021) [4] attempted to transfer the techniques of self-attention utilised by transformers in the paper "Attention Is All You Need" [12] which were developed primarily for use with text, to the domain of images. The basic methodology of a Vision Transformer is as follows:

1. Break-up the input image into sections, also referred to as patches (16x16 patches in the case of [4]).
2. Flatten these sections into vectors.
3. These flattened patches are then projected into a linear embedding space, and the resulting embedding vector is combined with a position embedding.
4. The combined embeddings are then fed as inputs to a transformer, which learns the representations of the patches combined with the positional encodings.
5. Typically the Vision Transformer is pre-trained on a large data-set of images (i.e ImageNet) and then fine-tuned to a smaller specific task by adding a relevant MLP head to the transformer to provide classification outputs.

The framework of the ViT is shown in Figure 2. It can be seen that the authors tried to maintain as similar a design as possible to the original transformers used for NLP, as originally shown in [12]. The ViT model has been shown to outperform ResNet models pre-trained on the same image-sets (ViT-H/14 achieving 88.55% accuracy on ImageNet, whilst ResNet achieved only 87.54%) while using significantly less computational resources.

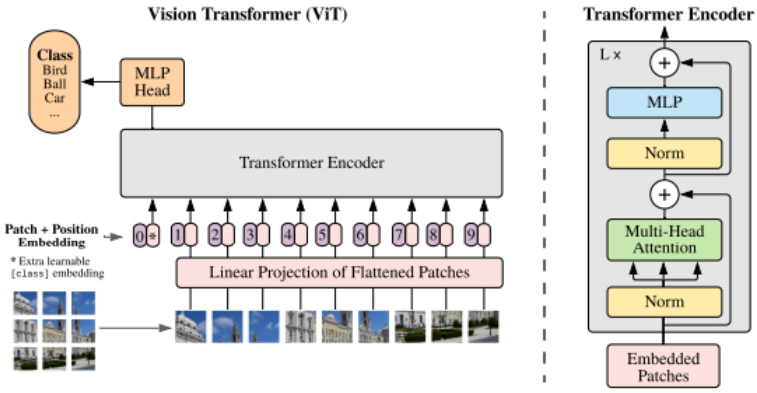


Figure 2: Framework of ViT [obtained from the original paper [4]]

However, there are a few limitations in using the ViT. The first is that they have to be fine-tuned on images that have the same resolution as the pre-training images, limiting the scope of potential uses of the model, as this can lead to information loss and give worse performance on specific downstream tasks such as object recognition and segmentation of images. Secondly, there is a quadratic increase in complexity as the image size increases, which can make models trained on higher resolution images intractable. To counter this problem, another group of researchers at Microsoft created a new vision transformer, using the ViT as groundwork, that can be fine-tuned on different image resolutions and scale in linear time, using the concept of shifted windows.

2.3.2 SWin Transformer

The difficulties in adapting from the language to the visual domain include the potential for variability in scale of objects in the images, and the higher resolution of image pixels when compared to words in text. The authors Z. Liu et al. developed the SWin transformer as a solution to these issues. SWin (Shifted Windows) Transformers [13] use a hierarchical structure to learn features of the images at different scales by merging patches in subsequent layers before being fed into a transformer. The hierarchical nature comes from the multiple layers of transformers. By applying self-attention to only the local windows, rather than the whole layer, the model scales linearly with image size. Similar to how CNNs use convolutions of different sizes, the merging of patches allows the transformers to learn more abstract representations of the image at different scales, solving the issue inherent in the ViT of lack of ability to recognise objects that have different scales in the image. The SWin transformer works according to these steps:

1. Break-up the input image into sections, also referred to as patches, of a predetermined size (authors use 4x4).

2. Flatten these sections into vectors.

3. These flattened patches are then projected into a linear embedding space of arbitrary dimension, and the resulting embedding vector is combined with a position embedding.

4. The combined embeddings are then fed as inputs to a transformer block. This is Stage 1.

5. The outputs of the Stage 1 are fed through a patch merging layer, concatenating features of each group of 2x2 neighbouring patches. Transformer blocks are then applied. This is Stage 2.

6. The above step is repeated twice more, as Stage 3 and Stage 4.

The diagram in Figure 3 demonstrates the above steps. The SWin subsequently performs better in general than the ViT on different benchmarks. For example, the author's tests gave an accuracy of 85.2% on the ImageNet22k for the ViT-L/16, while the Swin-L had an accuracy of 87.3%. In the subsequent sections, it will be shown that SWin Transformers allowed for significantly better model performance when compared to ViT on the MO106 data-set.

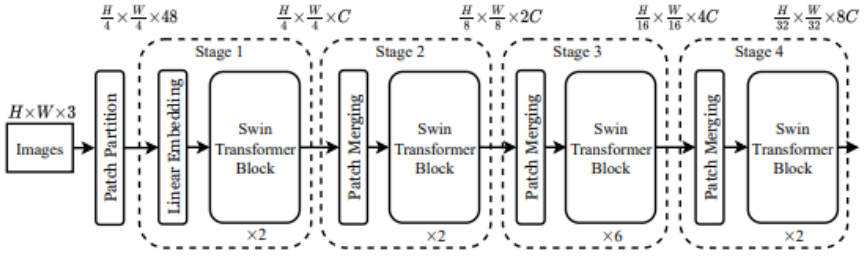


Figure 3: Framework of SWin Transformer [obtained from the original paper [13]]

2.3.3 ViTMAE

Another interesting development in the world of computer vision transformers is the ViTMAE, or the Vision Transformer Masked Auto-encoder. Produced by the authors K. He et al. in their paper "Masked Auto-encoders Are Scalable Vision Learners" (2022) [14], this is another algorithm that uses the ViT as a foundation for an improved algorithm. Whereas the previous papers utilised transformers in a supervised manner to encode the image into a lower dimensional latent space that captured the salient information contained within the images, the architecture within this paper utilises a masked auto-encoder combined with a lightweight decoder to train the transformer in a self-supervised manner. After pre-training the MAE-ViT (using the ViT as the encoder), the decoder can be removed and the ViT fine-tuned for image-tasks. Before the images are passed into the auto-encoder, they are broken up into patches, as with the previous transformers, but in this case a mask is applied to a majority of the patches (around 75%) in a uniformly random way. The encoder-decoder is then tasked with predicting the pixel values of the masked patches. By reducing the number of input patches to the encoder by 75%, the training time is reduced significantly. In steps, the model works as following:

1. Break-up the input image into sections, also referred to as patches, of a predetermined size.

2. Sample from these patches using uniform random sampling without replacement. Mask the remaining ones.

3. The unmasked patches are flattened and projected into a linear embedding space with the positional embeddings.

4. The projections are processed by the ViT transformer blocks.

5. The encoded output of the transformer blocks are input to the decoder, along with the mask tokens.

6. The MAE attempts to reconstruct the input by predicting pixel values for the masked patches, using a MSE loss function.

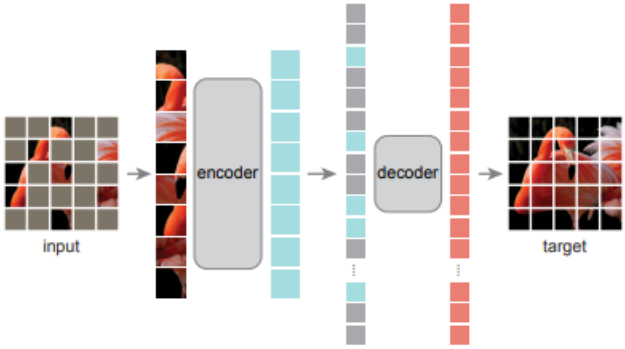


Figure 4: Simple diagram of ViTMAE [obtained from the original paper [14]]

The diagram in Figure 4 shows how the mask tokens aren't added to the model until after the encoder, which reduces the training time significantly. After pre-training, most likely on a large image set such as ImageNet1K, the encoder will have learnt a lower dimensional latent space of the images capturing salient information, and the decoder can be removed. For fine-tuning on a specific image task, the appropriate additional architecture can be added on top of the pre-trained ViT encoder (in the case of image classification, an MLP head/softmax layer can be added and the model fine-tuned to predict image categories). The performance of the ViTMAE is consistently good on different data-sets, as shown by the authors' own comparisons on the iNaturalist Data-set; their model consistently beats the current benchmarks. For example, on the iNaturalist2019 data-set, the best ViTMAE performance was 88.3% accuracy vs the previous best of 84.1%.
However, according to the authors, the true nature of how the masked model works is currently not understood. This is due to the fact the image masking is performed on random patches within the image which hold no semantic value (since they don't represent objects). Where masked auto-encoders in NLP learn to predict masked words which have their own semantic value, random patches of an image don't hold the same value. The authors state in their paper that therefore "our MAE infers complex, holistic reconstructions, suggesting

it has learned numerous visual concepts, i.e., semantics", and that it does this "by way of a rich hidden representation inside the MAE".

2.3.4 EffNet

This is the type of convolutional neural network that the authors in [9] used to create their benchmark model, which it is the aim of this project to improve upon. The EfficientNet model was created by the authors M. Tan et al. in their paper "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks" (2020) [3]. It attempted to improve upon previous efficient models such as MobileNet[15], which were created to meet a growing demand for models to run on embedded, mobile hardware. EffNet works via a scaling method, which scales all dimensions using a compound coefficient, while keeping the dimensions well balanced. The general idea is to fine-tune the model on a small problem, and scale the model up to a larger problem, while retaining the basic model structure and ratio of dimensions. The output of the authors' work is a family of EfficientNet models of varying sizes and complexity which can be applied to problems depending on computational resources and memory capacity.

3 Methodology

This dissertation intended to attempt the task of furthering the existing benchmarks relating to mushroom species image identification. The problem posed was worked against using a novel form of image recognition algorithm, called Vision Transformers. This study was conducted using quantitative research methods via machine learning algorithms performed on a high quality secondary source of data.
This methodology details the steps that were taken in carrying out this research, as well as the reasoning behind the decisions that were made. Firstly, the method of data collection will be covered, followed by how said data was utilised in the project, and the software used to work with the data. Lastly, the validity and reliability of the project will be discussed, whilst also covering the limitations of the project.

3.1 Data Collection

In order to meet the research problem posed, a high quality large data-set of mushroom images covering a broad number of species was required. Having performed a survey of the current literature on the topic of mushroom classification, a paper [9] was discovered which had created a data-set that could be easily utilised in other projects by downloading from their website https://keplab.mik.uni-pannon.hu/en/mo106eng. This data-set was curated from two primary sources of mushroom images: FGVCx Fungi Classification Challenge 2018 data-set and the Mushroom Observer website database. The FGVCx Fungi Classification Challenge 2018 data-set was made up of 85,578 training images and 4,182 validation images, covering 1394 different species of mushroom. The Mushroom Observer website was a collection of images taken by individuals and uploaded to the site, along with other information including location, name and certainty of identification label (on continuous scale from 1 to 3). A filtering criteria was applied to the images from the Mushroom Observer Website (MOW) so that only images of species with > 400 images and certainty ≥ 2 were included. Next, a classifier was built which would be able to distinguish full-shape mushroom images from less ideal images that include unnecessary features, and filtered both the FGVCx and MOW data-sets. More detail on this classifier can be obtained from the original paper [9]. The images left after this stage form the MO106 data-set, which is a collection of 29,100 images of 106 different species. An example of an image contained in the MO106 data-set is shown in Figure 5. This data-set fit the profile of the research question since it contained a large number of images, and covered a broad spectrum of species to be worthwhile. Also, this paper contained a benchmark by which to compare the results of this project.

While the authors in [9] took pains to ensure images were as clean as possible for training a classifier, there remain images within MO106 that contain distracting and useless elements, as shown by an example in Figure 6, which contains a book as the central object.

Figure 5: Image of Mycena leaiana [obtained from the MO106 dataset[9]]

3.2 Data Importing & Augmentation

The research conducted in this project relied on the use of Python and its subsequent modules relating to data processing and machine learning. The environment used to write the Python code was Google Colab due to the accessibility of their high-performance virtual GPUs to speed up data processing.
The image data-set downloaded came in a label-folder format. The function Split-folders was used to split the initial folder containing all the data into training and validation folders, following a 95%, 05% split. In order to then load these data-sets in an efficient way, the Torchvision function ImageFolder was utilised, since it recognises folder names as labels, and loads the images with the labels into a PIL format, and returns a dataloader object. This dataloader object can then be iterated to access each image. In this project, two dataloaders were instantiated, one for training and the other for validation. Whilst loading the image-data, transforms/augmentations were applied to the images individually via arguments in the ImageFolder function, to improve the generalisation capacity of the model. The transformations used were RandomResizedCrop, AugMix and Normalization in the training dataloader, and Resize and Normalization in the validation dataloader. Table 1 lists what each of these transformations do.

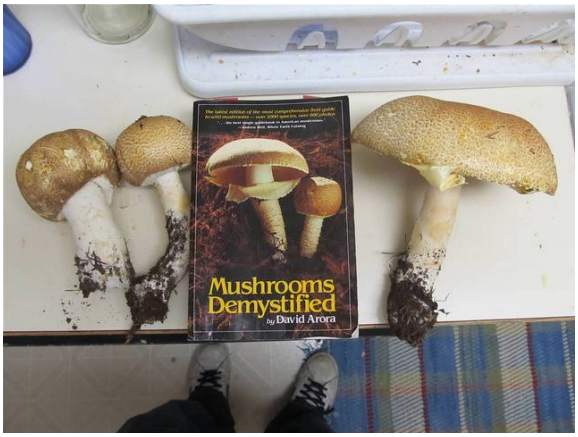


Figure 6: Image of Agaricus augustus with a book in the frame [obtained from the MO106 dataset[9]]

Transform	Description
Random Resize Crop	Crop a random portion of image and resize it to a given size
AugMiX [16]	Data augmentation method based on "AugMix: A Simple Data Processing Method to Improve Robustness and Uncertainty"
Normalization	Adjust all input channels to have mean of 0 and SD of 1
Resize	Resize the input image to the given size

Table 1: Transformation Descriptions

AugMix: Produced by the authors D. Hendrycks et al. at Google, this is a data processing technique that mixes randomly generated augmentations, utilising a Jenson-Shannon loss to ensure consistency. It has been shown to improve model performance on challenging image classification benchmarks.

3.3 Data Modelling

The Python module Huggingface[10], containing a number of useful machine learning algorithms and tools, was chosen to supply the Vision Transfomer algorithms. Specifically, the ones used were the 'google/vit-base-patch16-224-in21k', 'google/vit-base-patch32-224-in21k' and the 'microsoft/Swin-base-patch4-window7-224' pre-trained transformers, which were imported using the HuggingFace functionality, and the training arguments and metrics were also tuned and defined, where the metric would calculate the accuracy of the classifier in determining the species. The process of training and hyper-parameter tuning was then undertaken to find the best performing model. The final accuracy of the best-

performing model was then to be compared to the performance of the best model featured in the original paper [9] to demonstrate if the project had met the aims of the research, to improve the benchmark set by the previous paper.

3.3.1 ViT: 16x16 Patches, 224 Resolution

The first model to be fitted to the data was the standard Vision Transformer, which had been pre-trained on the ImageNet 21K data-set[20] - a collection of 14 million images from 21000 classes - and ready to perform fine-tuning. This transformer used 16x16 patches in its processing of the images, and was a variable that would be changed in order to discover the ideal patch-size. The following is a list of the parameters used during training:

Training Arguments	**Values**
Save Strategy	No
Eval. Stategy	Epoch
Learning Rate	2e-5
Per Device Train Batch Size	4
Per Device Eval Batch Size	4
Training Epochs	12
Weight Decay	0.01
Metric	Accuracy

Table 2: Training Arguments for ViT-16

In early runs of the experiment, a save strategy of every epoch was used, however this led to memory issues due to the limited amount of cloud storage available on Google Drive, so a different strategy was developed to overwrite the current best model if the next epoch resulted in an improved validation accuracy. The Huggingface package implementation of the transformers allows the model to be trained on GPUs if they are available, without having to specify explicitly in the code. This sped up model training by a significant amount. The model was chosen to have 16x16 patches, a good initial size since it was commonly used by the ViT creators in their paper [4]. Due to the ImageNet images having a resolution of 224x224, the ViT had to be fine-tuned on the same size images as it had been pre-trained on. Therefore any work performed with the ImageNet Pre-trained ViT had to have this size of image.
The model produced a running output of the validation accuracy and training/validation losses at each epoch. After the model was finished, it released a more detailed set of metrics for the final version of the model. This information was used to produce the training loss and accuracy graphs in Figure 7.
The best performance of this model was below the benchmark, so therefore it was decided to alter the size of the patches in the next implementation.

3.3.2 ViT: 32x32 Patches, 224 Resolution

It was decided to increase the size of the patches to 32x32 in this implementation, and analyse the performance for improvement. The same training strategy was kept as in the previous implementation, although the number of epochs was decreased to 10 in light of the lack of improvement in the previous model after a few epochs. The training output was used in producing the graphs shown in the results section in Figure 8.
Due to the inferior performance of this model compared to the previous attempt, it was decided that the use of standard ViTs was not a promising route to reach the benchmark, and therefore a different type of model was sought that might be able to offer more improvement. The SWin model was eventually seen to be more aligned with the task at hand, due to its higher degree of flexibility in image resolution and better generalisation capability.

3.3.3 SWin: 224 Resolution

It was decided to experiment with different implementations of the SWin algorithm, for the reasons mentioned above. Again, the Huggingface Python package provided the SWin model, which had been pre-trained on the ImageNet data-set. The default patch size of 4x4 and the following training arguments were used:

Training Arguments	**Values**
Remove Unused Columns	False
Eval. Stategy	Epoch
Learning Rate	5e-5
Gradient Accumulation Steps	2
Per Device Train Batch Size	16
Per Device Eval Batch Size	16
Training Epochs	10
Warmup Ratio	0.1
Logging Steps	10
Metric	Accuracy

Table 3: Training Arguments for SWin 224

There was a noticeable improvement in performance with this model, and the benchmark was surpassed. However, it was felt that this type of model had more capacity for improvement, due to the relative ease of the implementation. It was thought that one potential path to improving the model lay in increasing the resolution of images passed to the model. Since reducing the resolution causes potentially informative features of the image to be lost due to re-scaling, the general idea was to raise the resolution and allow the

transformer to search more deeply for patterns and discernible features at scales previously inaccessible.

3.3.4 SWin: 336 Resolution

The fine-tuning resolution of the model was raised to 336, with little changes elsewhere in order to note how the variable of resolution affects the model performance. Again, the Huggingface package was used, and the training arguments were kept the same as in the last implementation. An even bigger improvement was observed in training, which was felt to be more than sufficient in surpassing the benchmark behind this project, and setting a better benchmark to improve upon in future research. Therefore, no more modelling was performed in the project, and the results were analysed as recorded in the results section of this report.

3.4 Validity, Reliability & Limitations

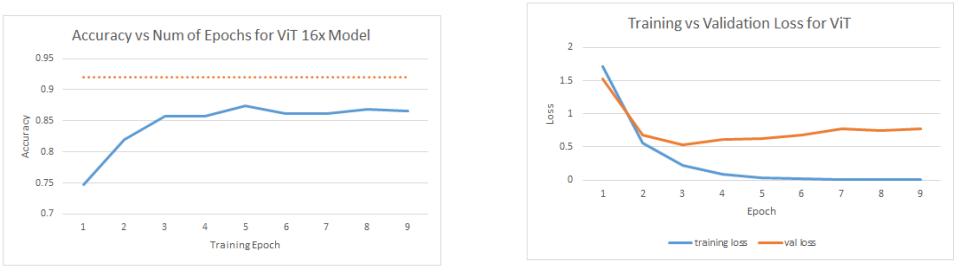
The tasks performed within this project are seen to be sufficient in answering the research question. The secondary data-set behind this project is verifiably from a source of integrity [9], and produced in a well detailed manner that was intended to improve the quality of images - and so validity of model - at the expense of species coverage and data-set size. The internal validity of the project is maintained by explicitly determining the independent variables in the study, while keeping the rest of the variables as stationary as possible. This is in order to reliably assess the effect the independent variables (Patch Size and Resolution) have on the models, while minimising the interference from the other variables. Randomisation is included in the study in the process of separating the initial data-set into a training and validation set, while maintaining a balanced distribution of species within each set. This means any model trained on the data-set will not suffer from biasing due to poorly balanced training and validation data. All models have been trained on the same training data, and evaluated on the same validation data, as to provide a direct comparison between models without the added noise caused by differences in data-sets, which could interfere with the model comparisons. The improvement of the benchmark on the MO106 data-set, as achieved by this project, is applicable to the wider sphere of vision modelling, as it demonstrates the potential that vision transformers have in shaking up the field of computer vision, whilst contributing a useful model to classify mushrooms for enthusiasts and professionals. Although the field of species that this project classifies is limited to 106, these species are common (due to the implicit filtering specification in the data-set design), and the resulting classification can help find the family of fungus that it belongs, and provide directions to potential areas of the mushroom world in which to search further. However, due to this same limitation in number of species, any application of this model to the real world of mushrooms cannot be taken as providing an accurate probability distribution of potential species, due to the sample species not representing the

full local & global population of species. In defense, this choice was taken as a compromise between model training validity and coverage, since if the data-set contained uncommon species, the number of images of these species will be limited, and the model will not learn to classify them well. This in turn would be detrimental to model performance.
Besides the limitations of the model mentioned previously, this project is limited in the amount of investigation that can be performed into the decision making process. This is due to the packaged nature of the algorithms in the HuggingFace module, and the novelty of the approach. Ultimately, the modelling performed in this paper is to obtain a discriminative probabilistic function that can map the input pixels into a conditional probability distribution of species (covered by model), given the image. A more scientific approach would be to model the joint probability distribution of the images and species labels, as more insight can be gained from this deeper understanding.

4 Results

4.1 ViT Model: 16x16 Patches, 224-Resolution

The first model to be trained was the 16x16-patch ViT model, which had been pre-trained on the ImageNet data-set. The training parameters were set to have 12 epochs, batch-size of 16, and gradient accumulation steps of 2. The model was trained for 9 of the 12 epochs, and stopped early due to a lack of improvement in results. The model was trained for 2 hours 42 mins before the early stopping. The results of training the 16-patch ViT are shown in the graph of validation accuracy in the left plot in Figure 7, with accuracy representing that of the model on the validation data. This plot also contains the benchmark that this project aims to improve upon. Also shown is the training curve which compares the training and validation losses over the course of the ViT model training, which can be seen in the right plot in Figure 7.

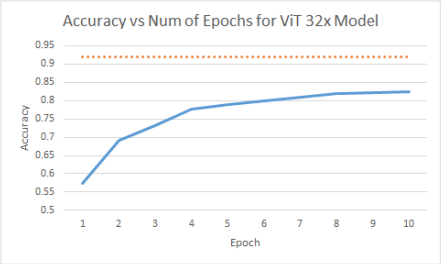


(a) Accuracy (b) Loss

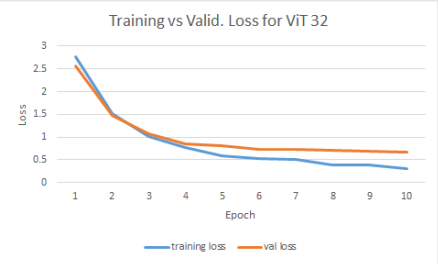
Figure 7: Graphs showing Validation Accuracy (l) and Training/Validation Loss (r) for the ViT 16xPatch 224-Res. Model

4.2 ViT Model: 32x32 Patches, 224-Resolution

The next model to be trained was the 32x32 Patch Size ViT, following the decision logic outlined in the methodology. The training arguments were kept the same, except for reducing the number of epochs to 10. The training output of this model produced the training graphs as seen in Figure 8. A comparison of the model validation accuracy can be compared to the benchmark in the left graph, while the training loss has been recorded in the right graph.



(a) Accuracy

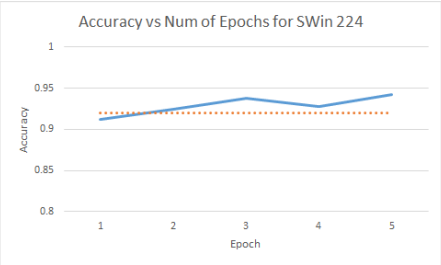


(b) Loss

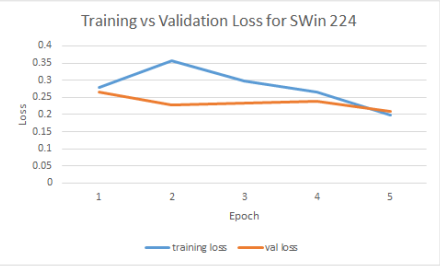
Figure 8: Graphs showing Validation Accuracy (l) and Training/Validation Loss (r) for the ViT 32xPatch 224-Res. Model

4.3 SWin Model: 224-Resolution

After receiving unpromising results from the ViT models, it was decided to find an alternative type of vision transformer that could offer more potential for improvement. Consequently, the SWin model was implemented in the project, with an initial image resolution of 224x224. The results of this model were output in a similar format to the ViT models - due to the consistency of HuggingFace - and recorded in the graphs as shown in Figure 9. This model training was initially set to have 10 epochs, however due to network interruptions and time constraints, the model training was stopped early. As the graph on the left shows, this model's performance surpassed the benchmark after only 2 epochs, which was a significant result.



(a) Accuracy

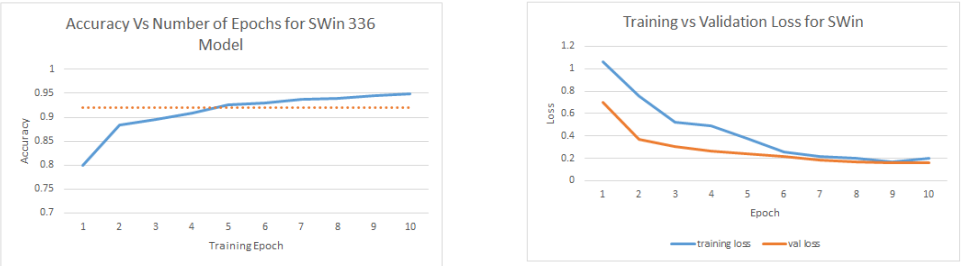


(b) Loss

Figure 9: Graphs showing Validation Accuracy (l) and Training/Validation Loss (r) for the SWin 224-Res. Model

4.4 SWin Model: 336-Resolution

The next step, the logic of which was outlined in the methodology section, was to increase the fine-tuning image resolution of the SWin Transformer to 336x336 and train on the same training data. The validation accuracy of this SWin model over training is shown in the left graph in Figure 10, while the curves of training and validation losses are shown in the right graph. The training time was 7 hours 32 mins, utilising the remote GPUs available via Google Colab. This graph shows that the model achieved a significant improvement over the benchmark, with a final accuracy of 94.9%, compared to 92.6%, proving the project to have been successful. The intricacies of what may have led to this improvement and successful conclusion to the project will be analysed in the following discussion section.



(a) Accuracy

(b) Loss

Figure 10: Graphs showing Validation Accuracy (l) and Training/Validation Loss (r) for the SWin 336-Res. Model

5 Discussion

5.1 ViT Models

This part of the discussion will focus on the performance of the ViT models and how they didn't meet the project benchmark.
Both ViT models had disappointing fine-tuning performance on the MO106 data-set. For the 16-patch model, the validation accuracy curve in Figure 7 (l) shows the model stopped improving after 3 epochs and performance plateaued, leading to the decision to terminate model training early to save on computational resources and time. The training curve in the right graph in Figure 7 shows how the model was over-fitting after the 3rd epoch, due to the training and validation loss curves diverging. As for the second ViT attempt with 32x32 patches, both graphs in Figure 8 show how training was smoother, but the model converged after about 5 epochs to an under-performing model instance, with a final accuracy of around 83%. The reasons for the stunted performance on this task could be due to the fixed architecture of the Vision Transformer[4][13]. The ViT does not allow for different sized patches during training, meaning only certain scales of images can be learnt in the process. However, the MO106 data-set, and the wider realm of mushroom imaging, contains variations in scale of the mushroom object within the image, due to the varying distances the photographer decides to take the image, and the amount of zoom. Perhaps the ViT model performance would have been superior had the data-set contained more consistent sizing of objects, but this is an unrealistic expectation of a collection of images. In summary, it was found that standard Vision Transformers were unable to meet the benchmark set by the authors Kiss and Czuni [9] of 92% validation accuracy on the MO106 data-set and therefore a new approach was required, which addressed the issues highlighted.

5.2 SWin Models

This part of the discussion will focus on the most significant results in the project, obtained by the SWin Transformers, and how the models surpassed the project benchmark.
The SWin Models produced the most promising results of this project. The first SWin transformer to be trained was the 224x224 resolution model. This had been pre-trained on the ImageNet data-set, same as with the ViT, and was fine-tuned on the MO106 data-set. The performance, as demonstrated by the validation accuracy on the left graph of Figure 9 and the Loss Curves on the right, show that the SWin model did not over-fit at any time during training, and the validation accuracy improved over training, up to a final accuracy of 94.2%. The accuracy curve in Figure 9 also shows that the accuracy improvement over subsequent epochs hadn't levelled off, giving the impression that, if run over more epochs, the accuracy could have improved further. This attempt had to be aborted after only 5 epochs due to training interruption and time-constraints, however the results obtained after only 5 epochs were very promising and had already surpassed the benchmark. It was thought that since this approach had already showed a significant performance gain

over the ViT, it was worth exploring this route further. The next model to be trained was altered to work with an image resolution of 336. This model trained in a smoother way to the previous attempt, and the training accuracy improved steadily until the last epoch. It surpassed the project benchmark after the fifth epoch, and due to the lack of convergence in the training curves, could have attained an even higher validation accuracy than 94.9% had it been set to train over more epochs. The unusually significant performance gains for the SWin over the ViT on this data-set could be related to how SWin transformers are hierarchical[13]. The SWin model has a merging layer embedded within the transformer, which acts to merge the patches after every stage in order to cover a larger area. This means, at each stage of the transformer, it is learning the salient features of the images at different scales, giving better generalisation performance and matching the real world expectations of a collated data-set of images gathered from different parts of the web.

Compared to the previous ViT model, the SWin models both had much better performance. In terms of best performance, the best SWin model had an accuracy of about 9% higher than the best ViT at the end of training. Additionally, the final accuracy of the SWin model of 94.9% is a significant improvement over the benchmark set by the authors Kiss & Czuni in their paper [9] of 92.6%.
This result meets the aims of the project to improve upon this benchmark, however the low number of mushroom species covered is an area that limits the applications of this project. In order to improve upon this result going forward, the same SWin model would be fine-tuned on a much larger, noisier data-set covering many more species, in order to allow such a model to be exhaustive of all viable species options, and image qualities, that it could be faced with in the real world.
Another limitation of using the HuggingFace module for this task is the packaged nature of the algorithm. Unlike CNNs, where feature-maps can be obtained to understand how the classifier has learnt the different features of the categories at different scales, HuggingFace do not allow the same for their algorithms to be picked apart and analysed to help understand the decision process. Therefore, the gains demonstrated in performance have to be balanced with the loss of interpretability of the classifier.
The results of this project therefore demonstrate how mushroom image classification rates could be significantly improved with the utilisation of the latest computer vision technologies, and also points to the potential improvements in other image domains via the application of vision transformers, which show promise in shaking up the world of computer vision.

6 Conclusion

Despite being an important area of study and application, it has been found that there is a lack of significant research in applying visual recognition algorithms to recognising mushroom species from images. Therefore, this project was intended to meet the requirement for more research emphasis in this area by applying cutting-edge computer vision techniques to the problem. By experimenting with different types of vision transformers on the MO106 data-set, this dissertation has found that the novel technology can perform better on the MO106 image-set of 106 mushroom species than a state of the art convolutional neural network representing the previous generation of image classification algorithms. This has been demonstrated by building a SWin model which surpassed the EffNet benchmark set by the authors in [9] of 92.6% with a final performance of 94.9% accuracy. By adopting new computer vision techniques, this project demonstrates not only that vision transformers are capable of surpassing the performance of traditional CNN architectures in image classification tasks on an example of a fine-grained image-set, but that vision transformers have the potential to shake up the whole realm of computer vision. In the course of the project, it was found that standard Vision Transformers (ViTs) were unable to meet the research aims due to an unscalable and inflexible architecture. However, a more scalable SWin transformer provided a more appropriate fit to the data and was able to meet the benchmark with appropriate image augmentations.

The ability to recognise mushrooms is considered a vital skill in many cultures, due to the purported variety of medicinal properties that many mushrooms contain. The lack of a wider cultural practice in embracing the benefits of mushrooms has led to a deficiency in transferable societal wisdom on the topic in many western cultures. This in turn has led these cultures to be fearful of utilising mushrooms due to the potentially deadly risks associated with them. In looking to rally against this alarming trend, this research project has set a new performance standard for classifying mushrooms at such a scale, and it will be worth investigating the further application of vision transformers to other fine-grained image recognition tasks.
Future research should be performed with similar modelling techniques on image-sets covering a wider range of mushroom species, which will help move towards a mushroom classifier that is generalisable and able to predict the correct mushroom with a high accuracy and confidence, meeting the long term goal of a practical algorithm that minimises the risk of misclassification. This will enable western cultures to engage with the mycological world with more confidence, and partake of their many underappreciated benefits.

References

[1] Yann LeCun & Yoshua Bengio; "Convolutional networks for images, speech, and time series." *The handbook of brain theory and neural networks 3361.10 (1995): 1995.*

[2] He, Kaiming & Zhang, Xiangyu & Ren, Shaoqing & Sun, Jian; "Deep Residual Learning for Image Recognition" (2016) 770-778. 10.1109/CVPR.2016.90.

[3] Mingxing Tan & Quoc Le; "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks" (2019) *Proceedings of the 36th International Conference on Machine Learning* PMLR 97:6105-6114, 2019.

[4] Dosovitskiy, Alexey; Beyer, Lucas; Kolesnikov, Alexander; Weissenborn, Dirk; Zhai, Xiaohua; Unterthiner, Thomas; Dehghani, Mostafa; Minderer, Matthias; Heigold, Georg; Gelly, Sylvain; Uszkoreit, Jakob; "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale" (2021) arXiv:2010.11929

[5] Sulc, Milan & Picek, Lukas & Matas, Jiri & Jeppesen, Thomas & Heilmann-Clausen, Jacob; "Fungi Recognition: A Practical Use Case" (2020). 2305-2313. 10.1109/WACV45572.2020.9093624.

[6] Preechasuk, Jitdumrong & Chaowalit, Orawan & Pensiri, Fuangfar & Visutsak, Porawat; "Image Analysis of Mushroom Types Classification by Convolution Neural Networks" (2019) 82-88. 10.1145/3375959.3375982.

[7] Ottom, Mohammad Ashraf & Alawad, Noor Aldeen; "Classification of Mushroom Fungi Using Machine Learning Techniques" (2019) *International Journal of Advanced Trends in Computer Science and Engineering.* 8. 2378-2385. 10.30534/ijatcse/2019/78852019.

[8] Z. Huang, J. Du & H. Zhang "A Multi-Stage Vision Transformer for Fine-grained Image Classification" (2021) *11th International Conference on Information Technology in Medicine and Education (ITME)* doi: 10.1109/ITME53901.2021.00047.

[9] Kiss, Norbert & Czuni, Laszlo "Mushroom Image Classification with CNNs: A Case-Study of Different Learning Strategies" (2021). 165-170. 10.1109/ISPA52656.2021.9552053.

[10] https://huggingface.co/docs/transformers/model_doc/vit
https://huggingface.co/docs/transformers/model_doc/swin

[11] Xie, Q., Luong, M. T., Hovy, E., & Le, Q. V. "Self-training with noisy student improves imagenet classification" In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 10687-10698).

[12] Vaswani, Ashish and Shazeer, Noam and Parmar, Niki and Uszkoreit, Jakob and Jones, Llion and Gomez, Aidan N and Kaiser, Lukasz and Polosukhin, Illia *Attention is All you Need (2017)* Advances in Neural Information Processing Systems

[13] A Liu, Ze; Lin, Yutong; Cao, Yue; Hu, Han; Wei, Yixuan; Zhang, Zheng *"Swin Transformer: Hierarchical Vision Transformer using Shifted Windows" (2021).* arXiv e-prints

[14] He, K., Chen, X., Xie, S., Li, Y., Dollár, P., & Girshick, R *"Masked autoencoders are scalable vision learners". (2022)* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 16000-16009).

[15] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, & L.-C. Chen *"MobileNetV2: Inverted residuals and linear bottlenecks". (2018)* Proceedings of the IEEE conference on computer vision and pattern recognition pp. 4510–4520

[16] Hendrycks, D., Mu, N., Cubuk, E. D., Zoph, B., Gilmer, J., & Lakshminarayanan, B *"Augmix: A simple data processing method to improve robustness and uncertainty" (2019).* arXiv preprint arXiv:1912.02781

[17] Cheung, P.C.K. *The nutritional and health benefits of mushrooms* (2010) Nutrition Bulletin, 35: 292-299 https://doi.org/10.1111/j.1467-3010.2010.01859.x

[18] Adenipekun, C. O.; Rasheedah Lawal *Uses of mushrooms in bioremediation: A review* Biotechnology and Molecular Biology Reviews 7.3 (2012): 62-68.

[19] Rootman, J.M; Kryskow, P.; Harvey, K.; et al. *Adults who microdose psychedelics report health related motivations and lower levels of anxiety and depression compared to non-microdosers* Sci Rep 11, 22479 (2021)

[20] J. Deng, W. Dong, R. Socher, L. -J. Li, Kai Li & Li Fei-Fei *"ImageNet: A large-scale hierarchical image database"* IEEE Conference on Computer Vision and Pattern Recognition doi: 10.1109/CVPR.2009.5206848