Evaluating the Localisation and Classification Performance of Deep Learning Architectures in the Context of Multi Pathology Chest X-Ray Classification.

Student Name: Joseph Crawley
Supervisor Name: Stamos Katsigiannis
Submitted as part of the degree of MEng Computer Science to the
Board of Examiners in the Department of Computer Sciences, Durham University

Abstract—Context/Background: Chest X-rays are invaluable diagnostic tools that offer a cost-effective method for identifying a range of pathologies. The application of image classification to Chest X-ray diagnostics has seen notable advancements due to deep learning technologies. These advancements have significantly improved classification accuracy, which identifies pathologies, and localisation accuracy, which determines the exact pathology locations. However, while classification accuracy has been widely studied, much of the literature still lacks comprehensive assessments of localisation accuracy, highlighting a gap in the existing literature.

Aims: The objective of this project is to explore the classification and localisation capabilities of various deep learning models with different architectures, specifically within the context of multi-label chest X-ray classification. Our research seeks to determine which state-of-the-art deep learning models provide the most effective classification and localisation performance when using transfer-learning techniques for this task. We specifically aim to fill a gap within the literature, evaluating localisation performance of the Swin transformer.

Method: We applied six deep learning models from the DenseNet, EfficientNet, and Swin transformer architectures to multi-label chest X-ray classification. These models were trained using the Chest-x-ray14 dataset and assessed on its official test split. Performance evaluation was conducted using the area under the curve (AUC) for classification accuracy and Intersection over Union (IoU) for localisation effectiveness. Model localisations were obtained through Grad-CAM, generating a heatmap localisation. We then created a demo application to showcase the aforementioned models.

Results: Five of our models consistently achieved an average AUC of 0.8 across all classes of the dataset. However, the Swin-Base model struggled to converge, resulting in a lower AUC of 0.68. Our DenseNet-169 model performs the best in the classification task with comparable results in many classes to the state-of-the-art. Notably, it achieves a higher Cardiomegaly AUC than previously reported in the literature, underscoring its superior performance in this specific category. Our localisation study yielded relatively poor IoU results compared to those in the literature, attributed by the use of heatmap localisation. From this study, we found DenseNet-169 to be the best performing model at localisation. Interestingly, although our EfficientNet-b0 model reached similar classification accuracies as others, it achieved a significantly lower average IoU. This discrepancy underscores potential differences in localisation effectiveness between models.

Conclusion: This project fulfilled its main objectives, we found our DenseNet-169 architecture provided superior classification and localisation performance compared to other models implemented within this project. We also addressed a gap in the literature by evaluating Swin Transformers localisation ability on Chest X-ray's. Future work could include the integration to localisation data for training, a further evalaution of the effectiveness of localisation methods, as well as further evaluation using multiple datasets.

Index Terms—Image Classification, Computer Vision, Artificial Intelligence, CXR, Deep Learning, Image classification

1 Introduction

The Chest X-ray (CXR) is one of the most powerful modalities clinicians can use as it is able to assist in diagnosis across a wide range of pathologies, with minimal expense. The current diagnostic system, which relies on human radiologists to interpret CXRs, faces significant bottlenecks without automated assistance. High volumes of CXRs demand substantial time for interpretation, leading to delays. According to the Association of American Medical Colleges (AAMC), the United States could face a shortfall of up to 124,000 physicians, including radiologists, by 2023 (AAMC 2021) [23]. This projection underscores the growing challenges in healthcare staffing which directly increases the workload for radiologists. Implementing tools which can assist in diagnostics not only alleviates the workload on radiologists, but also decreases the likelihood of scan misinterpretations caused by fatigue.

With the recent surge and advancements in the field

of deep learning, AI-powered tools have become prevalent in the field of medical imaging. Deep learning algorithms have demonstrated remarkable capabilities in enhancing diagnostic accuracy, processing speed, and overall efficiency in analysing medical images. For example, deep learning models have been successfully applied to detect anomalies in radiographic images, such as tumours in mammography scans [9].

A particularly notable study in this domain is the 2017 research by Rajpurkar et al. [26]. This model, known as CheXNet, was trained in the task of pneumonia detection from CXR input. The study's findings revealed that their model exceeded average human radiologist performance metrics, achieving a higher classification performance than human radiologists. This breakthrough not only highlights the potential of deep learning in medical diagnostics but also suggests a transformative shift in how such conditions can be diagnosed with greater reliability and less dependency on human evaluation.

Although the integration of deep learning into medical imaging has been a considerable success, as discussed by Zhang et al. [37], there are still concerns about its widespread adoption and usage. High classification accuracy of deep learning models can significantly enhance radiologist workflows [37]. However, these models will be of limited practical use unless their decision-making processes can be comprehended by human radiologists. Transparent and interpretable models are essential for ensuring that medical professionals can trust and effectively integrate AI insights into their diagnostic procedures.

The inherent lack of transparency in AI classification models is known as the black-box problem. It is crucial for methods to overcome the black-box problem to be implemented in image classification models used within radiology if these models are to be adopted in real clinical use. Overcoming this barrier is essential for ensuring that models can be trusted and utilised effectively by medical professionals in real-world diagnostic scenarios.

To address this, the field of explainable AI (XAI) seeks to create methodologies that can illuminate the reasoning behind AI decisions, thereby enhancing the transparency and usability of these technologies in healthcare environments. A common approach to introduce transparency into image-classifying models within Chest X-ray classification is Gradient-weighted Class Activation Mapping (Grad-CAM) [29]. Grad-CAM is a technique used to visualise the specific regions with an image that influences the decisions of an image classifier, thereby aiding in the interpretation of the model's decision.

The application of Grad-CAM is widely discussed within the literature, with notable examples including Wang et al.'s [34], and Hamza et al.'s [11] study on COVID-19 classification using Chest X-ray images, which integrates Grad-CAM visualisation into their classification model. Despite these advancements, the evaluation of localisation performance in Chest X-ray diagnosis remains unexplored. Although Wang et al. provide a quantitative analysis of the localisation capabilities of their model, their focus is limited to a single model architecture. Currently, there is a significant gap in comprehensive analyses of localisation performance across various deep-learning architectures in CXR diagnostics, highlighting a crucial area for research.

This project will apply state-of-the-art deep learning architectures from two major paradigms, Convolutional Neural Networks and Vision Transformers, to the task of multi-pathology Chest X-ray classification. An evaluation of model performance, as well as localisation performance, will then be conducted on each model, providing a comprehensive insight into which architectures perform the best in Chest X-ray image classification and localisation.

1.1 Project Purpose and Objectives

Research Question *Which state-of-the-art deep learning classification models yield the best classification and localisation performance in transfer-learning for multi-label Chest X-ray image classification ?*

This project aims to implement and evaluate current state-of-the-art deep learning models, in the task of multi-label CXR image classification via transfer learning. Evaluation will be based upon two primary components, classification performance and localisation performance. The conclusion to this project will be an analysis of which model architectures are optimally suited for CXR classification. Moreover, a quantitative evaluation of localisation performance for each model will be conducted, a novel contribution to the field.

1.2 Deliverables

The objectives of this project are split into three categories: basic, intermediate, and advanced:

Basic

- Identify suitable datasets that provide multi-labelled images of Chest X-rays.
- Create baseline deep learning models on the task of Chest X-ray classification with transfer learning.
- Evaluate baseline model performance.

Intermediate

- Implement localisation methods for classification models.
- Identify datasets with localisation of pathology data.
- Implement state-of-the-art CNNs to the task of Chest x-ray classification.
- Perform test set evaluation on classifier performance on model unseen images.

Advanced

- Implement state-of-the-art Vision transformers to the task of Chest X-ray classification.

- Perform a quantitative evaluation on model localisation performance.
- Implement a demo application to visualise trained models.

2 Related Work

2.1 Deep Learning for Image Classification

Image classification is a fundamental task in computer vision. Classification involves analysing visual content of an image and assigning it one or more labels. The advent of deep learning has revolutionised the field of image classification. This rapidly advancing technology, powered by deep neural networks, has demonstrated superior accuracy and efficiency in classifying and categorising visual input across a wide range of applications. Such applications include security [2], agriculture [19] and most commonly, medical imaging [31]. Convolutional Neural Networks (CNN's) [24] and more recently Vision Transformers [7] have emerged as pivotal architectures in image classification, transforming the analysis of visual data by extracting intricate features with exceptional accuracy and efficiency.

Although model architectures for deep learning in image classification (DLIC) vary significantly, the processes for model training and validation remain consistently similar. Two primary training procedures are predominantly used within DLIC: supervised and unsupervised learning. Additionally, semi-supervised learning, which integrates elements of both, also plays a crucial role in various development frameworks, although to a lesser extent. In supervised learning, we seek to train a model by minimising a loss function that measures the error between the predicted outputs and the actual labels. This can be formulated mathematically as:

$$\hat{\theta} = \arg\min_{\theta} L(\theta; \mathbf{X}, \mathbf{y})$$

where:

- $\mathbf{X}$ is the matrix of input features, each row representing an input example. For RGB images with height H and width W, $\mathbf{X}$ is a $(3, W, H)$ matrix.
- $\mathbf{y}$ is the vector of labels, each element corresponding to the label of the corresponding input in $\mathbf{X}$.
- L is the loss function, a measure of the prediction error.
- θ are the parameters of the model, adjusted to minimize L.

In unsupervised learning, models independently discover underlying patterns and structures in the data without relying on pre-assigned labels. This approach is commonly employed in tasks such as clustering, where the model groups similar data points together, or in image generation, where it creates new images that mimic the characteristics of the input data.

Deep Learning has transformed image classification with its ability to process and categorise large amounts of visual data. However, this technology faces several challenges and limitations. Primarily, these models require high-quality datasets that are costly and time-consuming to assemble, and any flaws, such as biases within datasets can cause minority-represented groups to be treated unfairly. This can be seen in Buolamwini et al's work [5], where authors expose severe accuracy disparities within minority gender and ethnic groups, highlighting a clear bias in image recognition systems using 'Pilot Parliament's benchmark'. However, this bias is substantiated by systemic inequalities in society and is a wider problem outside of the scope of machine learning.

Another key limitation to image classification within deep learning is the 'black-box' problem, where model decision-making is unclear. The reasoning behind a classification is integral to the usage of such a system. In many applications, understanding the rationale behind a classification is crucial to the system's utility and trustworthiness. This transparency is essential not only for validating the outcomes, but also for ensuring compliance with regulatory standards, and facilitating user acceptance. To address this challenge, the concept of explainability in AI models has become increasingly important.

2.2 Deep Learning Image Classification Architectures

As previously discussed, deep learning has revolutionised the field of image classification, offering unprecedented accuracy and efficiency. Given the rapid advancements in this area, it is crucial to stay informed about the current state-of-the-art technologies. At the time of writing, the two leading architectures driving these advancements in image classification are 'Convolutional Neural Networks' (CNNs) and 'Vision Transformers' (ViTs). The following sections provide a concise overview of the historical development and key milestones achieved by these two architectures.

2.2.1 Convolutional Neural Networks (CNNs)

"Convolutional Neural Networks" (CNNs) were first introduced by LeCun et al. in their seminal 1989 paper titled "Backpropagation Applied to Handwritten Zip Code Recognition". In this study, the researchers trained a multi-layer convolutional neural network on the task of handwriting recognition. This work was pivotal in demonstrating the power of CNN and paved the way for Krizhevsky et al.'s 'AlexNet' [17]. The development of 'AlexNet' marked a significant milestone in the evolution of CNNs. This model, which was trained using modern GPUs and featured several innovative techniques such as ReLU activation functions, dropout, and data augmentation, dramatically improved the performance of CNNs in large-scale image recognition tasks. AlexNet not only won the ImageNet competition by a large margin, but also set new standards in image classification, reaffirming the powerful capabilities of deep convolutional networks and catalysing a surge in AI research focused on deep learning.

Since the inception of models such as 'AlexNet', there has been significant improvements in the architectural design of CNN's, yielding new and improved models and demonstrating excellent performance in image classification tasks.

Models such as ResNet [12], DenseNet [14] and EfficientNet [32] have demonstrated the power of CNN's setting benchmarks in image classification.
DenseNet (Densely Connected Convolutional Network) introduces a novel architecture where each layer is directly connected to every other layer in a feed-forward fashion. This 'dense' connectivity ensures maximal information flow between layers, making the network more parameter efficient and resulting in, at the time (2015), state-of-the-art ImageNet classification accuracy with fewer parameters then ResNet (previous state-of-the-art).
EfficientNet, on the other hand, scales up CNNs systematically through a compound coefficient that balances the depth, width and resolution of the network, leading to improved accuracy and efficiency. This model again set, at the time (2019), state-of-the-art ImageNet classification accuracy.

A typical CNN architecture for image classification includes several layers: convolutional layers, activation layers (ReLU), pooling layers, fully connected layers, and a final output layer with an activation function. In the context of Chest X-ray classification, these layers work together to detect indicative features of pathologies presented within an image of a Chest X-ray.

Descriptions of the above layers are as follows:

- **Convolutional Layer:** Applies numerous filters to the input to create a feature map that summarises the presence of detected features in the input. For instance, a convolutional layer might identify edges, and another one might identify more complex patterns in an image.
- **Activation Layer (ReLU):** Introduces non-linearity into the network, allowing the network to learn complex patterns.
- **Pooling Layer:** Reduces the dimensionality of each feature map but retains the most essential information.
- **Fully Connected Layer:** Computes the class scores that result in image classifications, Each neuron in this layer will be connected to previous layers.
- **Output Layer:** Provides probabilities for each class, making it useful for classification among multiple classes.

2.2.2 Vision Transformers

Vision Transformers (ViTs) were first introduced by Dosovitskiy et al. in their groundbreaking 2020 paper 'An Image is worth 16x16 words: Transformers for Image Recognition at Scale" [8]. In this study, the researchers applied the transformer architecture, which had previous success in natural language processing (NLP), to the task of image classification. Treating image patches as a conventional transformer would treat words. ViTs use self-attention mechanisms to process global dependencies between patches, allowing the model to weigh the importance of different parts of images contextually.

The development of ViTs marked a significant paradigm shift in the field of computer vision, traditionally dominated by CNNs. Unlike CNNs, which inherently capture local features due to their convolutional nature, transformers focus on both local and global image features. This architecture dramatically improved performance on various image recognition benchmarks including ImageNet, challenging the supremacy of CNNs and opening new avenues for research in architectures that combine the strengths of both CNNs and ViTs (hybrid architectures).

Since the inception of the original ViT model, there has been significant advancements and research into Vision Transformers.
A notable advancement is the development of the Swin-Transformer [21]. Unlike the original Vision Transformer which processes the whole image in terms of global self-attention, the Swin Transformer divides the image into non-overlapping local windows. Inside each window, self-attention is computed, which significantly reduces computational complexity. The complexity of a traditional ViT with patch size $\mathbf{n}$ is $O(n^2)$, whereas in a Swin Transformer it is $O(n)$ [30]. This method not only improves efficiency but also enables the model to maintain a comprehensive understanding of the entire image, proving advantageous in image classification tasks.

High level descriptions of ViT architectures are as follows:

- **Input and Patch Embedding Layer:** The input layer is split into fixed-size patches, which are then flattened and linearly transformed into a sequence of embeddings. These embeddings serve as the input tokens for the transformer.
- **Positional Encoding:** Adds position information to the patch embeddings to retain the order of the patches.
- **Transformer Encoder:**
  - **Mult-Head Self-Attention:** Allows the model to jointly attend to information from different representation subspaces at different positions, enhancing its ability to focus on various parts of the image.
  - **Feed-Forward Networks:** Consist of two linear transformations with a non-linear activation in between, processing the output of the attention mechanism.
- **Normalisation Layers:** Typically applied before each multi-head attention and feed-forward network block in the encoder, and after adding the positional encodings.
- **Classification Head:** After processing through the transformer encoder, the output corresponding to a special classification token ('CLS') is used to predict the class of the input image through a linear layer.

2.3 Transfer Learning

Transfer learning is a technique used within machine learning where a model developed for one task is reused and 'fine-tuned' for a separate, similar, and often more specific task. This technique is based on the fundamental

idea that knowledge gained from learning one task is relevant to a new task and thus can improve learning performance on a new task. Transfer learning, first introduced by S. Bozinovski [39] in the early 1980s, has revolutionised the field of deep learning, allowing powerful models to be created with limited access to large datasets.
In the realm of image classification, transfer learning is leveraged to yield impressive models in a multitude of tasks. This increase in the use and efficacy of transfer learning for image classification is largely due to datasets such as 'ImageNet' [6].
ImageNet is a dataset with over 14,000,000 labelled images, serving as a benchmark for several tasks in computer vision. The inception of the ImageNet dataset led to the launch of the yearly 'ImageNet Large Scale Visual Recognition Challenge' (ILSVRC) [28]. Each year, participants leverage models pre-trained on ImageNet to innovate and refine approaches to deep learning, furthering the state-of-the-art in many computer vision tasks.

Furthermore, transfer learning facilitates a more accessible approach to machine learning. It lowers the barrier to entry for those with less computational power or smaller datasets. By fine-tuning pre-trained models, researchers and developers can achieve competitive results without the need to train large models from scratch.

In essence, transfer learning, especially in the realm of image classification, has revolutionised the way models are developed. It allows for efficient use of resources, promotes rapid innovation and continues to influence a wide range of applications beyond just computer vision, including areas such as natural language processing and predictive analytics.

2.4 Chest X-ray Classification literature

Automation of the CXR diagnostic workflow has been a significant field of research, gaining momentum in the latter half of the 20th century. Early endeavours, such as those documented by Becker et al. [4], employed classical algorithms to automate aspects of image analysis, yielding promising results that marked initial success in the field.
The landscape of automation significantly evolved with the advent of machine learning techniques in the late 20th century, which introduced more sophisticated analytical capabilities. The integration of deep learning over the past decade has further revolutionised this workflow, enhancing accuracy and efficiency through advanced feature extraction and learning capabilities from large datasets.

2.4.1 Datasets for Image Classification of Chest X-rays

There are various datasets publicly available for Image classification of Chest X-rays, however for this project to fulfil its aims, the chosen dataset must possess particular characteristics. This section compares four prominent datasets: the NIH Chest-x-ray14 Dataset [34], the 'CheXpert' [15] dataset from Stanford University [15], the MIMIC-CXR dataset [16]. Each dataset is evaluated based on the following key characteristics essential for our project's success:

- **Dataset Size:** The dataset must be sufficiently large enough to facilitate the training of deep classification models.
- **Image and Label Requirements:** The dataset must contain images coupled with labels which span multiple pathologies.
- **Localisation Data:** The dataset must have some form of localisation data to allow for a quantitative evaluation of model localisation performance.
- **Literature Relevance:** The dataset must be used in prior studies to enable direct comparisons with other research.

The CheXpert dataset contains 224,316 CXR images from over 65,240 patients. [15]. While this dataset is sufficiently large to train deep learning models and spans a wide range of pathologies, the dataset presents several limitations. One significant issue is label ambiguity, which arises from its reliance on automated text-based methods to assign labels to imaging data. This NLP-driven process can introduce errors and inconsistencies due to the subjective nature of radiological interpretations found in medical reports, potentially leading to inconsistent training outcomes and challenges in model evaluation. Another limitation is the limited external validation of the dataset. CheXpert primarily comprises images from Stanford Hospital, reflecting a relatively homogenous patient population in terms of demographics and health profiles. This lack of diversity can hinder the generalisability of models trained on this dataset to other populations or settings, which may have different prevalences of diseases and diagnostic practices. Additionally, the absence of localisation data in CheXpert limits its use in applications, requiring detailed anatomical localisation of pathologies, such as precise disease marker identification and advanced diagnostic tasks. The dataset lacks annotations like bounding boxes, which are crucial in the quantitative evaluation of model localisation performance.

The MIMIC-CXR dataset [16] comprises a robust collection of 371,920 CXRs. Each image is paired with a corresponding radiologist's report; however, these reports lack predefined binary labels. Consequently, additional preprocessing efforts are required to extract binary labels from the textual reports, a necessary step for utilising this dataset in the training of image classification models.

The NIH Chest-x-ray14 dataset [34] is one of the most extensively used datasets for image classification in Chest X-rays. Cited by over 2000 studies, it serves as a benchmark for state-of-the-art image classification models. Comprised of 112,120 labelled images, it includes 14 pathology labels alongside a 'no finding' label. Additionally, a valuable subset of this dataset contains localisation data, with bounding boxes provided by radiologists to pinpoint the locations of pathologies. Although this is not applied to all 112,120 images, there are 983 localisation images with bounding box data spanning 8 pathologies, making it more

than sufficient for a localisation evaluation.

Table 1 gives a brief overview of the surveyed datasets.

TABLE 1
Overview of datasets

Dataset	Size	Binary Labels	Localisation	Cited by
Chest-x-ray14	112,120	Yes	By bounding box	4191
MIMIC-CXR	371,920	No	No	669
CheXpert	224,316	Yes	No	2279

2.4.2 State-of-the-art Chest X-ray classification

To achieve the primary objectives of this project, which include implementing CXR image classification models and conducting a thorough assessment of both classification and localisation performance, it is essential to perform a comprehensive literature review. This review will establish the current state-of-the-art in Chest X-ray classification and assess the localisation evaluation methods used in related studies. The surveyed literature will be classification models which are trained using the Chest-x-ray14 dataset as this is the most prevalent within the literature (see Table 1).

Kufel et al.'s study, 'Multi-Label Classification of Chest X-ray Abnormalities Using Transfer Learning Techniques' [18], employs the EfficientNet architecture as the backbone for a multi-label classifier applied to the Chest-x-ray14 dataset. This recent work, published in September 2023, achieved state-of-the-art classification performance, with an average AUC of 0.843 across various pathologies. While this study asserts that it achieves state-of-the-art classification performance on the Chest-x-ray14 dataset, I argue that these claims may not be valid. The authors deviated from the official training and testing splits provided by the dataset, which compromises the comparability of their results. This study employs visualisation methods, in the form of heatmaps, to support model localisation; however, there is no quantitative evaluation of localisation performance using the Chest-x-ray14's bounding box data.

In their study 'Weakly Supervised Deep Learning for Thoracic Disease Classification and Localization on Chest X-rays,' Yan et al. [36] utilise a DenseNet-based CNN for multi-label classification on the Chest-x-ray14 dataset. They achieved an average validation AUC of 0.8302 across all 14 pathologies using the official test split; this performance is currently unmatched, marking it as the state-of-the-art for this dataset.
The title of this work emphasises localisation, yet the authors employ visualisation techniques in an ablation study, which constitutes a qualitative rather than quantitative approach. Notably, their evaluation lacks a metric-based assessment of localisation performance. Moreover, the study is limited to just eight selectively chosen images. Given the critical importance of accurate localisation in medical diagnostics, I contend that this is an inadequate evaluation of the model's localisation capabilities.

Gundel et al.'s study, 'Learning to Recognize Abnormalities in Chest X-Rays with Location-Aware Dense Networks' [10], extends the application of DenseNet models to Chest X-ray classification. Unlike other studies previously discussed, this research utilises both the Chest-x-ray14 and the PLCO [38] datasets, the latter being a notable cancer study resource containing Chest X-rays. They report AUC scores of 0.807 on the official Chest-x-ray14 split and 0.841 using a custom split.

The success in Chest X-ray classification extends beyond studies utilizing CNNs. For instance, Taslimi et al. in their work 'SwinCheX: Multi-label Classification on Chest X-ray Images with Transformers' [33], deploy a Swin Transformer [21] trained on the NIH Chest-x-ray14 dataset. Their model, 'SwinCheX,' achieves an AUC score of 0.810, comparable to state-of-the-art models. This achievement underscores the potential of transformer-based architectures in improving Chest X-ray classification outcomes. Authors leverage transfer-learning on a pre-trained Swin-Large transformer with weights initialised from ImageNet [6].
However, while this study is significant in demonstrating the value of transformer architectures in CXR classification, it provides a limited evaluation of localisation performance, primarily illustrated through a brief application of Grad-CAM.

The seminal work by Wang et al. which introduced the Chest-x-ray14 dataset [34] not only detailed methodology on the creation of this important dataset, but also trained the first DLIC model on this dataset. When this work was made, the Chest-x-ray14 dataset was known as Chest-x-ray8, as the introduction of 6 new labels was a later endeavour. In this work, Wang et al. implement 4 different pre-trained models, AlexNet, GoogleLeNet, VGGNet-16 and ResNet-50 and leverage transfer learning to train on their newly devised dataset. They achieve an average AUC across 8 pathologies of 0.7891. This study also includes a localisation analysis using model-generated heatmaps. The authors quantitatively assess localisation performance by calculating the intersection over the bounding box of these heatmaps, although this method is applied solely to the ResNet-50 model.

The current state-of-the-art in pathology localisation on the Chest-x-ray14 dataset is presented by Rozenberg et al. in their study, 'Localization with Limited Annotation of Chest X-rays' [27]. The authors adopt the 'PreActResNet' [13], a variant of the ResNet architecture [12], which incorporates an activation layer prior to each weight layer, enhancing feature extraction for localisation tasks. They introduce a novel loss function that leverages bounding box data from the dataset to refine localisation accuracy. Their model segments the Chest X-ray into patches, assigning probabilities for the presence of pathologies in each, thus generating a heatmap. This approach differs from model-agnostic techniques like Grad-CAM [29], which can generate heatmaps without specific model adaptations. The authors of this paper produce state-of-the-art localisation

results in the form of an IoU metric, the average IoU across 8 pathologies within the localisation set is 0.79.

Another work which focuses on the localisation task of CXR classification is Li et al.'s work 'Thoracic Disease Identification and Localisation with Limited Supervision' [20]. Authors follow a similar approach to Rozenberg et al.'s [27] previously mentioned work, also implementing preact-ResNet [13] model for localisation. However Rozenberg et al. critique this approach, pointing out that it potentially undermines the accuracy of the localisation, as evidenced by a comparatively lower IoU score. The reported IoU score from Li et al.s work was measured over several thresholds. Threshold values are utilised to convert a heatmap into a localisation map. Specifically, a threshold value of 0.5 indicates that a localisation consists of regions where pixel intensities exceed 127, on a scale from 0 to 255. In this work, 7 thresholds are used 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7 and the respected average IoU's for each threshold are as follows: 0.73, 0.62, 0.48, 0.37, 0.28, 0.20, 0.12.

The comprehensive literature review conducted in this study has elucidated the current state-of-the-art in CXR image classification. Notably, the work of Yan et al. [36] emerged as a significant contribution, demonstrating a model that achieved a classification AUC of 0.8302 on the official Chest-x-ray14 split. Furthermore, the survey highlighted the potential of Vision Transformers in CXR classification, as evidenced by Taslimi et al.'s [33] findings that Swin Transformer models can achieve classification accuracies comparable to the state-of-the-art. The state-of-the-art localisation performance was also found in Rozenberg et al's study [27] with their CNN-based architecture.

However, the review also identified a gap in the existing literature. While state-of-the-art classification architectures, such as the Swin transformer [21], have been implemented as classifiers [33], there have been no quantitative evaluation of these architectures localisation performance. This gap raises the question - can Swin transformers provide an improved localisation over CNNs?

Table 2 has a brief overview of the above-surveyed literature.

2.5 Evaluating Image Classification

In the realm of medical diagnostics, the accuracy and reliability of image classification systems are paramount. When evaluating such a system, it is crucial to employ robust metrics that comprehensively assess not only the classification accuracy, but also the precision of localisation.

2.5.1 AUC - Area Under Reciever Operating Characteristic Curve

The Area Under Curve (AUC) is a widely used performance metric in machine learning, which can be used to evaluate a range of architecturally different models in classification performance. The metric itself is the area under the receiver operating characteristic Curve (ROC), The ROC curve plots the true positive rate of a classifier (TPR) or recall against the false positive rate of a classifier (FPR) or (1 - Inverse Recall). The AUC metric when calculated is between 0 and 1, with 1 being a perfect classifier, 0 being the worst possible classifier, and 0.5 being as good as randomly guessing.

While AUC is typically applied to binary classifiers where the notion of false positive and true positive is clearly defined, it can be adapted for use in a multi-label setting using the One vs Rest (OvR) approach. This approach treats each label as a separate binary classification problem and calculates the AUC for each label independently and an average of each label can be taken.

AUC is particularly effective for a plethora of reasons. One of which is its simplicity, a simple higher is better scale is effective for comparing classifier performance. Additionally, AUC is a threshold invariant as the FPR and TPR are measured across all possible thresholds, making it a robust metric for classifiers in imbalanced datasets.

The efficacy of AUC as a metric is further reinforced by its widespread adoption in image classification of CXRs, where its utility is well-documented. Studies like those by Kufel et al. [18], Pillai et al. [25] and Baltruschat et al. [3] illustrate AUC's critical role in evaluating classifier performance accurately in medical imaging, demonstrating its relevance and reliability in practical Chest X-ray image classification environments.

2.5.2 IoU - Intersection Over Union

In the critical context of Chest X-ray classification, evaluating an image classifier's performance necessitates robust metrics for both classification accuracy and localisation precision. This dual requirement ensures that the classifier not only identifies the presence of medical conditions accurately, but also pinpoints their specific locations within the X-ray images. This is crucial for effective diagnosis, as well as enhancing the trust radiologists place in classification models.

IoU is calculated by dividing the area of overlap between two the predicted localisation and the ground truth localisation by the area of their union. Mathematically this is formulated as:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

IoU provides a clear, quantitative measure of how accurately a model predicts the location of classifications. A quantitative measure allows for direct standardised comparisons between models.
Although Intersection over Union (IoU) is widely used in studies focused on localisation [20], [27], it is notably underused in Chest X-ray image classification. Many studies neglect to assess localisation performance [10], [18],

TABLE 2
Surveyed Multi-label Chest X-ray classification Papers

Authors	Localisation Or Classification	Validation Performance	Model Architecture	Localisation	SOTA
Kufel et al. [18]	Classification	AUC: 0.843 (Custom Split)	EfficientNet (CNN)	Grad-CAM no Evaluation	No
Rozenberg et al. [27]	Localisation	IoU: 0.79	preact-ResNet(CNN)	Loss function for localisation	
Yan et al. [36]	Classification	AUC: 0.8302 (Official Split)	DenseNet (CNN)	Grad-CAM no Evaluation	Yes
Li et al. [20]	Localisation	IoU: 0.73	preact-ResNet(CNN)	Loss function for localisation	
Gundel et al. [10]	Classification	AUC: 0.807 (Offiical Split)	DenseNet (CNN)	None	No
Taslimi et al. [33]	Classification	AUC: 0.810 (Official Split)	Swin Transformer (ViT)	Grad-CAM no Evaluation	No
Wang et al. [34]	Classification	AUC: 0.789 (Official Split 8 pathologies)	DenseNet (CNN)	Grad-CAM with Evaluation	No

[33], [36]. Considering the critical role of model localisation in the application of image classification tools for Chest X-ray diagnostics, the adoption of a quantitative metric like IoU to evaluate localisation effectiveness is crucial.

3 Solution

3.1 Programming Language and Framework Selection

Python, a standard programming language for machine learning research, was chosen for the development of this project due to its extensive libraries to assist in the development and evaluation of machine learning models . The Pytorch machine-learning framework was chosen due to its extensive support for a variety of deep-learning tasks.

Table 3 contains the list of Python modules used within the project:

TABLE 3
Python Modules used

Module	Description
Pillow (PIL)	Image processing
scikit-learn	Provides evaluation metric (AUC)
Pytorch	Extensive library for deep learning
torchvision (Pytorch)	Extends pytorch for computer vision
matplotlib	Used to create graphs
timm	Used to obtain pretrained models
gradio	Facilitates the GUI for demo application
pytorch gradcam	Grad-CAM for pytorch models

3.2 The NIH Chest-x-ray14 Dataset

When selecting a dataset to meet the objectives of this project, several criteria must be satisfied: the dataset should be sufficiently large to enable effective training, include labels for a diverse set of pathologies, be commonly utilised in the literature to ensure comparability, and contain localisation data, potentially in the form of bounding boxes. The NIH Chest-x-ray14 [34] fulfils the requirements detailed in section 2.4.1, it is used in a wide range of studies, containing 112,120 images across 14 different pathologies. This dataset also contains 985 images with radiologist-marked bounding boxes, making it sufficient to evaluate the localisation performance of models.

The Chest-x-ray14 dataset contains an official train and test split. This official split has been critiqued by Kufel et al. (2023) [18], where authors raise concern, stating 'an average 3 times more photos per patient compared to the training split'. While these concerns are valid, potentially leading to a decrease in variance in the test set, to ensure this work is comparable to others in the literature [33], [36], [10], the official split was used.

The NIH Chest-x-ray14 dataset also contains a wide range of non-image features including patient age and gender. These non-image features can, and have been, in related works used in an attempt to facilitate an increase in classification performance, with works such as Baltruschat et al. in 'Comparison of Deep Learning Approaches for Multi-Label Chest X-ray Classification' [3] utilising non-image features and observing an increase in classification performance.

Integrating non-image features is anticipated to enhance performance, mirroring a real radiologist's approach, which typically extends beyond evaluating CXRs to reach a diagnosis. However, this integration can also introduce bias into the decision-making model. For instance, using gender as an input feature could reinforce prejudiced interpretations within medical treatments and diagnostics, resulting in skewed model predictions that disproportionately impact the accuracy for certain genders. To avoid these risks and uphold the integrity of our predictions, this project will deliberately exclude non-image features from the input dataset.

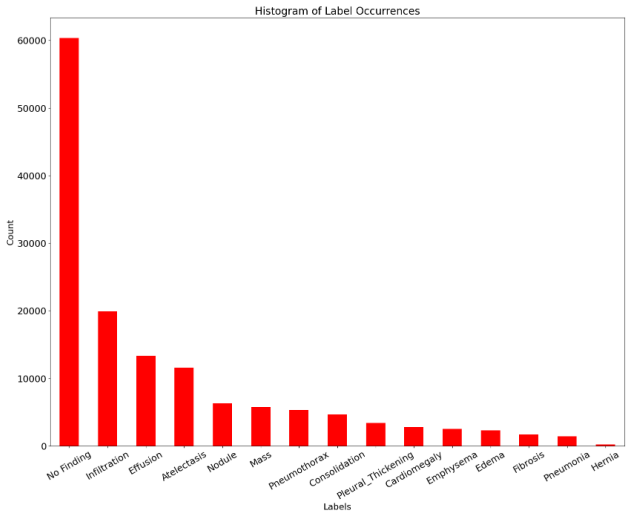

Fig. 1. Histogram of Label distribution.

Figure 1 shows the frequencies of labels within the dataset; this dataset suffers from severe class imbalance with severe underrepresentation of fibrosis, pneumonia and hernia. Conversely, there is also an overrepresentation of instances labelled as 'No-Finding'.
Given these disparities, strategic decisions must be made in the training and evaluation phases of the model development. These decisions are crucial to mitigate risks of overfitting and ensure that the imbalanced class distribution does not skew evaluations.

3.3 Methodlogy Design Choices

Data augmentation is a critical technique employed to enhance the robustness and accuracy of models. By introducing minor variations to training images, data augmentation improves a model's ability to generalise to unseen data, thereby improving accuracy and reducing overfitting. The various data augmentations implemented in this project are random horizontal flips as well as a random rotation of a maximum of 15 degrees. These augmentations are standard within deep learning for image classification and can be seen within a wide range of related literature. [3] [25] [18] The operations mentioned above are randomly applied to training images with half chance to create additional training images, thereby enhancing the dataset's variability.

The images within the dataset are 1024x1024 greyscale images. Because this project is utilising transfer learning, further augmentations will be applied to ensure models are optimally configured to process these input images.
Firstly, the images are resized to 224×224. This size is standard for ImageNet pre-trained classifier models. These models also expect 3 colour channels in input (RGB), and therefore the native greyscale images are converted to RGB. Models used in transfer learning are typically trained on images that have been normalised to fit specific colour distribution profiles. By applying the same normalisation methods during the fine-tuning process, the model can achieve faster convergence. This increased efficiency is because the training data closely mimics the distribution of the pre-trained model's original data. Aligning these distributions allows the model to leverage its pre-acquired knowledge more effectively, optimising learning speed and enhancing performance.

The normalisation values can be seen below, these are standard for models pre-trained on ImageNet.

Let μ_r, μ_g and μ_b be the mean colour values for red green and blue respectively; with σ_r, σ_g and σ_b being each standard deviation.

$$\mu_r = 0.485, \quad \mu_g = 0.456, \quad \mu_b = 0.406$$

$$\sigma_r = 0.229, \quad , \sigma_g = 0.224, \quad \sigma_b = 0.225$$

The previously mentioned transforms are all implemented through pytorch's 'torchvision' library using the 'transforms' class.

A further measure to mitigate overfitting on class imbalance is to define the over-reperesented 'no finding' label as the absence of other labels. This is beneficial as it reduces the complexity of the classifiers by reducing the number of classes by one, and also shifts a model's focus towards detecting specific pathologies, which I hypothesise will increase diagnostic strength. This method of encoding the no-finding label as the absence of other labels is also used heavily within the literature [18], [36], [10], [33].

3.3.1 *Loss Function and Hyperparameter selection*

This project aims to develop multi-label image classifiers in Chest X-ray classification, therefore an appropriate loss function must be used. Unlike a multi-class classifier, where each image belongs to exactly one class, in the multi-label approach an image can belong to more than one class. This means that traditional loss functions, such as categorical cross-entropy loss cannot be used.

A traditional multi-label classifier will be trained using Binary Cross Entropy loss (BCE). However, this loss function is sensitive to class imbalance, and as the Chest-x-ray14 dataset suffers from class imbalance, this is impractical. Therefore, we followed the approach of Baltruschat et al. [3] and implemented a class-averaged BCE loss, mitigating the class imbalance problem.

The binary Cross-Entropy loss for a single observation, where $y \in \{0, 1\}$ is the true label and $\hat{y}$ is the predicted probability, is given by:

$$\text{BCE} = -\left(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right)$$

Class averaged Binary Cross-Entropy extends traditional BCE by taking the average across each class c where c has N_c observations. This is mathematically formulated as follows:

$$\text{Class-Averaged BCE} = \frac{1}{c}\sum_{c=1}^{C}\left(\frac{1}{N_c}\sum_{i=1}^{N_c}\text{BCE}(y_i^{(c)}, \hat{y}_i^{(c)})\right)$$

Class-averaged BCE mitigates class imbalance issues at the cost of complexity traditional BCE is computed in constant $O(1)$ where Class-Averaged BCE has a linear complexity of $O(n)$ where n is the number of classes.

To achieve a fair evaluation of all the models implemented in this project, a single learning rate will be applied to each model. Although each model would have its own optimal learning rate, standardising the learning rate allows for consistent comparisons. Through experimental selection, an initial learning rate of 0.0005 is used, and the optimiser chosen is an Adam optimiser.

3.4 Method Formulation

The image classification transfer learning task can be formulated as such: Dataset D is split into three subsets D_{Tr}, D_{Val} and D_{Te} the train, validation and test splits respectively. D_{Te} is the official test split provided by the Chest-x-ray14 dataset [34], located in (`test_list.txt`). D_{Tr}, D_{Val} are subsets of the official training and validation split, found in (`train_val.txt`). These subsets are divided into the ratio 1:9 respectively, which is done in a stratified fashion to ensure fair class representation in both D_{Tr} and D_Val. A further dataset we call D_{Loc} (The localisation dataset) is devised using the localisation data provided in the Chest-x-ray14 dataset.

Dataloaders will be created for each subset to allow their iteration for training and testing. These DataLoaders utilise PyTorch and its multiprocessing features, assigning four cores to each DataLoader to optimise the use of computational resources, thus accelerating training and evaluation processes. The DataLoaders are constructed using PyTorch Dataset objects; consequently, a custom Dataset class, DataSetHelper, interfaces with a PyTorch Dataset to facilitate this setup. Within this class, the dataset file is processed, obtaining transformed and resized images as a $3 \times 224 \times 224$ tensor. The labels are a 14 tensor with 1's in positions, indicating a positive instance of a pathology and zeros elsewhere.

All of the models chosen will be pre-trained on ImageNet in the interest of fair comparison, and will be of the following architectures: DenseNet-121 [14], DenseNet-169 [14], EfficientNet-b0 [32], EfficientNet-b1 [32], Swin-Small [21] and Swin-Base [21]. As this project is leveraging transfer learning, each of these architectures will be initialised with pre-trained weights from ImageNet. These pre-trained models were obtained through 'timm' [35], a Python package. Each model will be trained for a maximum of 30 epochs (30 iterations through the training data). After each epoch, the model performance will be evaluated on the validation dataset, and the loss and AUC will be calculated. If the validation loss has not decreased for 3 epochs, the learning rate will be divided by 2, a method used in [33]. If there is no decrease after 5 epochs, the model will stop training, in the interest of reducing overfitting and computational time. If the model's validation loss has decreased, the model weights will be saved. After training has terminated the model weights with the lowest validation loss will be loaded and the D_{Te} dataloader will be iterated through obtaining test results and AUC will be recorded.

Algorithm 1 Model Training Procedure

Let D_{train} be the training dataset
Initialise model parameters θ
repeat
  Sample batch $X = (X_{\text{i}}, y_{\text{i}})$ from D_{train}
  for each (X_i, y_i) in X **do**
    $\hat{y} \leftarrow f_\theta(X_{\text{i}})$
    Calculate loss $L(\hat{y}, y_i)$
  end for
  Update model parameters θ, with adam optimiser

Following model training and testing, the D_{Loc} subset will be used to perform the quantitative localisation evaluation of each model. We iterate through each image and bounding box of D_{Loc} and generate a heatmap visualisation from a model, this is done through Grad-CAM. Following this, we will generate an IoU metric for each image pertaining to each class, which will give us a per-class IoU metric. Section 2.5.2 shows a basic implementation of IoU. Here, I will extend this to the specific mathematical process used in this project.

A model generated heatmap is a $224 \times 224 \times 1$ image which can be reperesented as a matrix. $H = [a_{ij}]_{224\times224}$ with $a_{ij} \in [0, 255]$. We first apply a function: $T(x, a)$, where x is an element and a is a threshold value, to each element within H to yield a binary mask with 1s indication the localisation region. $T(x, a)$ is defined as follows:

$$T(x, a) = \begin{cases} 1 & \text{if } x > a \cdot 255, \\ 0 & \text{otherwise.} \end{cases}$$

Let A be the resulting matrix after $T(x, a)$ has been applied to the model generated heatmap and B be the matrix where each element is set to 1 if it corresponds to a location within the bounding box provided by D_{Loc} and 0 otherwise. Both A and B are 224×224 matrices. Then IoU can be defined as follows:

$$\text{IoU}(A, B) = \frac{(A \circ B)}{(\min(A + B, 1))}$$

Here, $\circ$ denotes the Hadamard product (element-wise multiplication) representing the intersection, and $\min(A + B, 1)$ ensures that all entries in the matrix sum are capped at 1 for union calculation, effectively mapping any sum of 2 to 1 and everything else to 0.

The training validation and testing procedures for both classification and localisation outlined in this section are implemented through one class `ModelProcess(model)`. In the constructor of this class, a model is initialised with its pre-trained ImageNet weights. The `TrainOfficialSplit` method initialises D_{Tr} D_{Val} and D_{Te} dataloaders with the necessary transforms detailed in section 3.3. Training is conducted using the `TrainOfficial` method, which invokes the `ValidateMultiLabel` method after each training epoch to assess validation results and save model weights as needed. Upon completing the training, the `TestResultsMultiLabel` method is called to evaluate the models performance on D_{Te} and save these results.

The `TrainOfficial` method conducts the training procedure and calls upon `ValidateMultiLabel` after each epoch of training. To obtain validation results and to save the model weights if required, and once training has finished, the `TestResultsMultiLabel` function is called, which generates the models performance on the test set.

After all the models were trained and evaluated based on their classification and localisation performance, a demo application was developed. This application is designed to showcase the capabilities of each model included in the project, featuring both their localisation and classification outputs. To enhance user-friendliness, the application was built with a graphical user interface (GUI).

To facilitate this, the 'gradio' [1] Python library was used. This library provides an easy to setup GUI with methods designed for the integration of deep learning models. This application allows a user to choose a classification model, upload an image of a CXR, and then output a localisation as well as the class predictions.

3.4.1 Grad-CAM

Grad-CAM, short for Gradient-weighted Class Activation Mapping, first introduced by Selvaraju et al. [29], is a technique used to gain insights into which parts of a given input image were influential for a classification model when making a decision. This is particularly useful for visualising the regions within an image that lead to certain classifications. Grad-CAM is a localisation method, originally designed for use on CNNs by visualising the final convolutional layer, thus yielding a ROI for model decision-making. This process is extended for use within Swin transformers [21] by visualising a final self attention map.

Here is a Mathematical formulation of Grad-CAM, inspired by the work of Selvaraju et al. [29]:

1) Compute the gradient of the target class y^c with respect to feature maps A of a convolutional layer for CNN's or self-attention mechanism of a Swin transformer.
$$\frac{\partial y^c}{\partial A_{ij}}$$
where i and j index the spatial location in the feature map.
2) Global-average-pool these gradients across the spatial dimensions (i, j) to obtain the neuron importance weights α_k^c
$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}$$
where k indexes the channel dimension of the feature maps and Z is the total number of pixels in the feature map.
3) Calculate the Grad-CAM activation map $L^c_{\text{Grad-CAM}}$ by performing a weighted combination of forward activation maps, followed by a ReLU to only retain features with a positive influence on the class of interest:
$$L^c_{\text{Grad-CAM}} = \text{ReLU}\left(\sum_k \alpha_k^c A^k\right)$$

3.5 Model Architectures

3.5.1 EfficientNet

The EfficientNet image classifiers, first introduced by Tan [32], are available in eight variants from b0 to b7. For this project, we will employ two of the smaller models: EfficientNet b0 and b1. This selection was informed by several factors. Primarily, the limited size of our dataset poses a challenge; larger EfficientNet models are likely to struggle with convergence and may overfit, given the data constraints. Furthermore, the increased input resolution required by larger models could introduce variability in performance comparison across different architectures. Higher resolutions can artificially enhance performance, obscuring the intrinsic effectiveness of the model designs. This project aims to maintain consistent input resolutions to ensure that differences in performance are attributable to architectural distinctions rather than input scale.

EfficientNet-b0 is built for an input resolution of 224x224x3, while the b1 variant is built for 240x240x3. EfficientNet-b1 can still handle input resolutions of 224x224x3 and therefore it is sufficient for a fixed input resolution of 224x224x3.

In this project, we will use EfficientNet-b0 and EfficientNet-b1. These models will be pre-trained with ImageNet weights and obtained through 'timm'.

An architectural overview of EfficientNet-b0 can be seen in Figure 2. [35].

3.5.2 DenseNet

DenseNet, first introduced by Huang et al. [14], is a deep neural network architecture that distinctively connects each layer to every other layer in a feed-forward fashion. The architecture is renowned for its efficiency in parameter use.

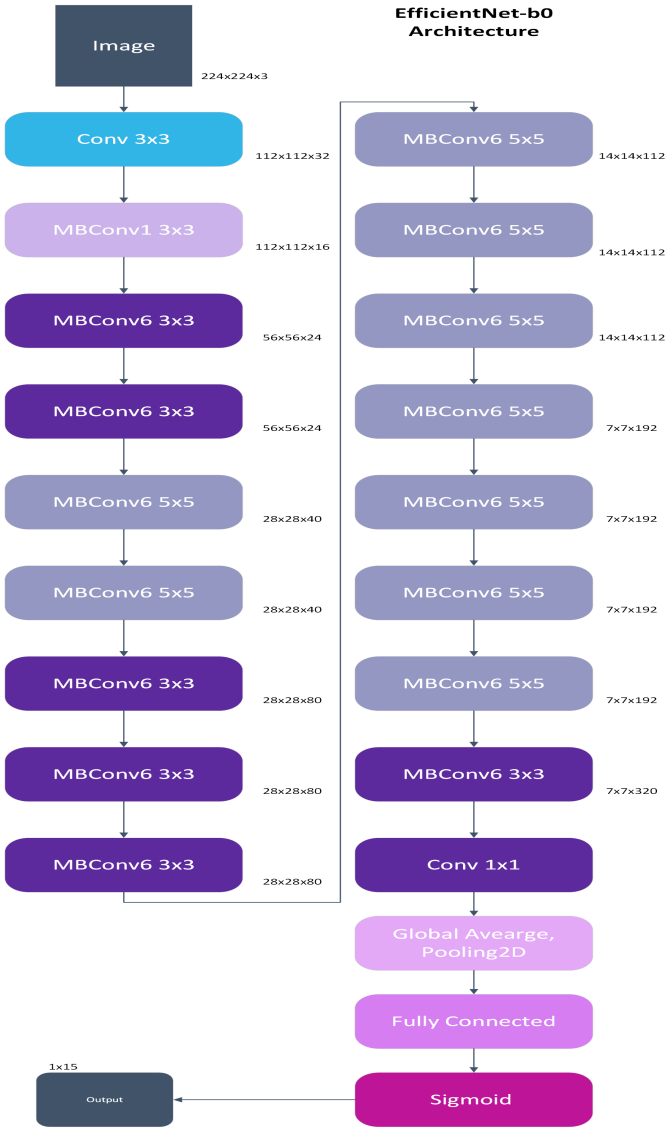

Fig. 2. EfficientNet-b0 architecture adapted from original implementation with the introduction of a sigmoid activation function, as to allow multi-label classification. Based on the description of model architecture in [32]

This efficiency is largely due to feature reuse within each layer.

Each DenseNet model is made up of 4 'denseblocks', which are made up of several dense layers (depending on the specific DenseNet model). Each layer within a denseblock is connected to all subsequent layers.

Mathematically, the output of each layer x_l in a DenseNet is computed as:

$$x_l = H_l([x_0, x_1, ..., x_{l-1}])$$

where $[x_0, x_1, ..., x_{l-1}]$ represents the concatenation of the feature outputs of all preceding layers, H_l is a composite function consisting of the following operations:

- **Batch Normalisation:** This function normalised the inputs of each layer using the mean and variance of the batch data. It adjusts and scales activations to improve the stability and performance of the neural network [14].
- **ReLU (Rectified Linear Unit):** This activation function introduced non-linearity into the model's learning process, which is essential for learning complex patterns. It is defined mathematically as:

$$ReLU(x) = max(0, x)$$

- **Convolutional Operation:** In DenseNet, this is a 3x3 convolution [14], which manipulates the input data using a filter W to produce feature maps. This operation is defined as:

$$Conv(x) = W * x + b$$

Where, W represents the weight matrix of the filters, $*$ denotes the convolution operation, and b is a bias term added to each output of the convolution.

Each denseblock within a DenseNet model contains $[a, b, c, d]$ dense layers. For the two DeneNet models used within this project (DenseNet-121 and DenseNet-169) a, b, c, d are $6, 12, 24, 16$ and $6, 12, 32, 32$ respectively [14].

The DenseNet models used within this project will have initialised weights from ImageNet, and will be obtained through 'timm' [35].

3.5.3 Swin

In this project, we incorporate two Swin Transformer models: Swin-Small and Swin-Base [21]. These models are larger than the CNN architectures previously discussed. The decision to opt for these larger Swin models was influenced by the demonstrated efficacy of the Swin-Large transformer in the task of Chest X-ray classification [33]. Although implementing the Swin-Large model would have been a viable option, its considerable size led us to select the two smaller variants instead. This strategic choice balances the need for computational efficiency with the desire to leverage the advanced capabilities of Swin Transformers.

Here is a brief overview of how Swin Transformers work. This is based upon the original Swin Transformer paper [21]:

Swin Transformers work by dividing an input image into small patches. The size of the patches for Swin models we use are 4×4, these patches are then linearly transformed into token embeddings that serve as the input to the transformer layers. This initial step effectively turns the image into a sequence of tokens, a method used for words in NLP and adopted for images in [8].

The architecture processes these tokens through multiple layers that operate at different scales (see figure 3), gradually reducing the resolution and increasing the feature dimension. This hierarchical approach allows the model to capture global features of an image.

A key feature of Swin Transformers is the use of shifted window partitioning for computing self-attention. In each

Transformer layer, self-attention is computed within non-overlapping local windows. The windows are shifted in subsequent layers, alternating between the original and a shifted configuration. This approach restricts the self-attention to local image areas, while the shifting mechanism ensures that the model can integrate information across the entire image. The feature allows a Swin Transformer to effectively capture local and global features of an image.

The basic components of a Swin Transformer Layer are as follows:

- **Layer Normalisation:** Each Swin Transformer layer starts with Layer Normalisation, which is applied before other operations like self-attention and MLP blocks. Layer normalisation helps stabilise the training process and is a similar method to Batch Normalisation (see 3.5.2).
- **Shifted Window Attention:** Swin Transformers make use of 'shifted window multihead-self-attention' this component is split into two distinct parts:
  - **Window MSA:** In the first phase, self-attention is computed within local windows (e.g., dividing the image into 4x4 patches). These fixed-size windows restrict the area over which self-attention is computed, thereby reducing computational complexity and allowing the model to capture local features.
  - **Shifted Window MSA:** In the next layer, the window configuration is shifted to traverse the image. This shift overlaps adjacent windows from the previous layer, allowing for cross-window interaction and helping the model to capture global features.

  Layer normalisation and Shifted Window Multihead Self-attention alternate through the network, ensuring coverage of features across the image.

- **Multilayer Perceptron (MLP):** Following the self-attention phase, each layer includes an MLP block, which consists of two dense layers with GELU activation function in between. This activation function introduces non-linearity, allowing the Swin Transformer to learn a wider range of patterns in an image.

3.5.4 Model implementation

As stated previously, all of these models are implemented through 'timm' [35]. This package facilitates the use of pre-trained models with weights. Specific timm implementation details are in Table 4. Each of these models has ImageNet fine-tuned weights. A linear layer is applied onto the end of each model, which has input features of the final output layer. This is then connected to a sigmoid function to obtain multi-label outputs.

3.5.5 Localisation Implementation

For localising model features, we employ the `pytorch_gradcam` package, which offers a variety of classes designed to facilitate the implementation of Grad-CAM across numerous deep-learning models. In the case of our (CNNs), DenseNet and EfficientNet models, we utilise their final convolutional layers to produce gradients for Grad-CAM. The relevant layers are identified as `model.features[-1]` for DenseNet and `model.conv_head` for EfficientNet models. On the other hand, for our Swin Transformer models—which differ from CNNs in architecture—we apply Grad-CAM to the final self-attention block, specifically `model.layers[-1].blocks[-1].norm1`.

TABLE 4
List of Models and their names in Timm

Model Name	Timm Name
DenseNet-121	`densenet121.tv_in1k`
DenseNet-169	`DenseNet169.tv_in1k`
EfficientNet-b0	`efficientnet_b0.ra_in1k`
EfficientNet-b1	`efficientnet_b1.ft_in1k`
Swin-Small	`swin_small_patch4_window7_224.ms_in1k`
Swin-Base	`swin_base_patch4_window7_224.ms_in1k`

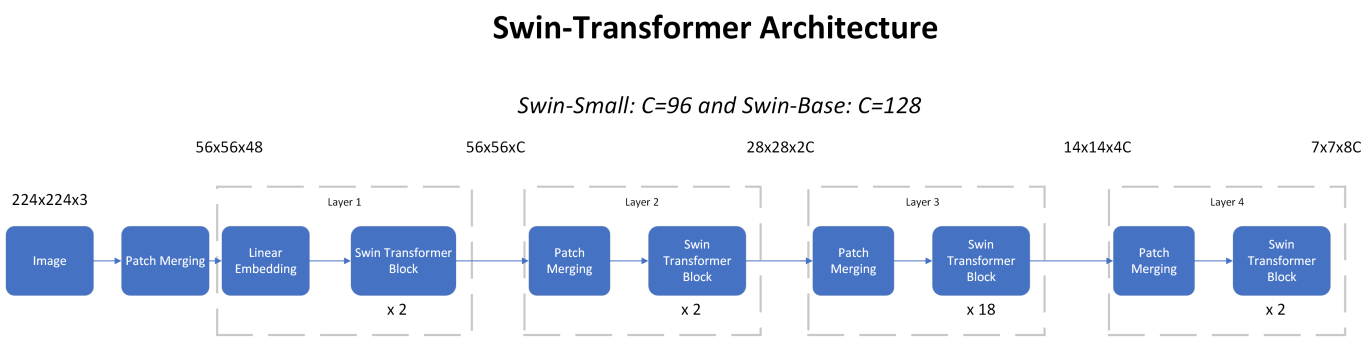

Fig. 3. Swin-Transformer hierarchical architecture inspired by [21]

4 Results

4.1 Evaluation Metrics

In evaluating the classifiers developed for CXR classification tasks, we primarily employ two criteria, following precedents set in existing literature. The first criterion is the Area Under the Curve (AUC) of the Receiver Operating Characteristics (ROC). We selected AUC due to its prevalent use in previous studies, see section 2.4.2, which facilitates direct performance comparisons across different research. Moreover, AUC is advantageous as it is independent of any decision threshold, thereby providing a holistic measure of model accuracy under various conditions.

In the context of multi-label classification, the concept of a 'threshold' becomes pivotal. Thresholds in this context convert a model's continuous probability outputs into definitive binary outcomes (e.g., setting a threshold at 0.5 classifies probabilities above this value as positive). However, employing a static threshold can be problematic, particularly with datasets like Chest-x-ray14 [34], which features 14 distinct pathologies in an imbalanced distribution. Such a fixed threshold approach may inadvertently bias the model towards more frequently represented classes, such as 'Infiltration.' Therefore, a threshold independent evaluation metric such as AUC is required to ensure fair and accurate classifier evaluation.

In this project. to allow AUC to be used in the multi-label instance we employ the One vs Rest approach (see section 2.5.1). This produces a per class AUC, allowing us to evaluate the specific classifications of each pathology. We will then take a macro average of each class AUC to yield an average AUC which is insensitive to class imbalance.

To measure model localisation performance, we will employ an IoU metric, as seen in section 2.5.2. We will follow the procedure outlined in section 3.4 and obtain the IoU at thresholds T(0.1),T(0.15),T(0.25),T(0.35) and T(0.45). These threshold values are chosen as this range of thresholds produced the best results in the surveyed work, specifically Li et al's work [20].

4.2 Results and Analysis

4.2.1 Training Results

Figure 6 and 7 show the validation loss of each model per epoch. DenseNet-121, DenseNet-169, EfficientNet-b0, EfficientNet-b1, Swin-Small and Swin-Base were trained for 18,14,11,11,27 and 15 Epochs respectively. Early stopping was triggered for each model due to no improvement of best validation loss in 5 epochs. The validation loss plots show a sharp initial decline, which levels off in later epochs, indicative of overfitting.

Among the six loss curves presented, each model achieves an average minimum validation loss of 0.134, except for the Swin-Base model, which ends training with a significantly higher loss. This discrepancy could likely be attributed to the choice of learning rate. Although a constant learning rate of 0.0005 was applied uniformly across all models—proving effective for most—it appears not to have been optimal for Swin-Base, leading to its relatively poorer performance in reducing validation loss.

4.2.2 Classificaiton Results

Tables 5 and 6 display the AUC performance of our trained models on the official test split of the Chest-x-ray14 dataset. In these tables, 'pneumothorax' and 'pneumonia' are abbreviated as 'pneu1' and 'pneu2,' respectively. Each model consistently achieves an AUC of 0.8 across all classes, with the exception of the Swin-Base model, which records a 0.68 AUC.

The Chest-x-ray14 dataset spans 14 pathologies. These have been abbreviated in the results tables and are as follows: Atelectasis, Cardiomegaly, Effusion, Infiltration, Mass, Nodule, Pneumonia, Pneumophorax, Consolidation, Edema, Emphysema, Fibrosis, Pleaural Thickening and Hernia.

Our models achieved their best performance in the following pathologies: Edema, Cardiomegaly, and Hernia, with an average AUC of 0.88 across all models, excluding Swin-Small. In contrast, our models consistently underperformed on Atelectasis, Consolidation, and Infiltration. This trend of underperformance in these

categories is also evident in the surveyed literature, as documented in several studies [10], [18], [33], [34], [36]

We benchmarked our top-performing model, DenseNet-169, against other published results on the same dataset to ensure a fair comparison. Notably, our DenseNet-169 model outperformed all competitors in the Cardiomegaly class and demonstrated near state-of-the-art AUC scores in several other categories.

4.2.3 Localisation Results

Table 7 presents the localisation results for each model across various thresholds, revealing some intriguing findings. The overall Intersection over Union (IoU) scores detailed in this table are notably low, particularly when compared to the IoU scores reported by Wang et al. [34]. Their best average IoU was 0.63, compared to ours, 0.133. This discrepancy primarily stems from the differences in the methods used for model localisation. In our approach, localisations are represented as heatmaps, which can assume any geometric shape based on the model's output, whereas the localisations provided by the radiologists in the dataset are standardised bounding boxes (rectangles). Consequently, achieving high IoU scores with our heatmaps is challenging because they are unlikely to conform to the rectangular shapes necessary for high IoU comparisons. In contrast, Wang et al. [34] convert heatmaps into bounding boxes before calculating IoU, a technique that yields significantly better results than the methodology employed in this project.

Although our localisation results do not match those reported in the literature, they remain comparable within our study. Interestingly, despite EfficientNet and DenseNet models achieving similar AUCs in classification performance, the EfficientNet-b0 model report a considerably lower IoU than our DenseNet models. This suggests differences in model architecture may significantly influence localisation performance, despite comparable classification performance.

Following low IoU results on Grad-CAM visualisation for our models, we experiment by using EigenCAM [22]. This method is similar to Grad-CAM as it generates a heatmap of localisation. However, this technique relies on a different computational strategy that focuses on aggregating feature importance across layers, rather than leveraging gradient information from the last convolutional layer alone. This allows for a more comprehensive visualisation that may capture subtler aspects of model reasoning across the entire network architecture [22]. EigenCAM localisation generation is significantly more computationally demanding than Grad-CAM, which prevented us from obtaining EigenCAM results for each threshold, unlike our comprehensive data with Grad-CAM. Table 8 illustrates the findings from our model localisation study using EigenCAM at a threshold of 0.1. The data indicates that the average Intersection over Union (IoU) has improved for all models except for Swin-Base. Although these results are noteworthy, the absence of data across multiple thresholds restricts our ability to definitively determine which localisation method is better for this task.

4.3 Demo Application

The implementation of our Demo Application is within two programs: `DemoForLocalisation.py` and `DemoForClassification.py`. A screenshot of `DemoForLocalisation.py` can be seen in the Figure 4, and a screenshot of `DemoForClassification.py` can be seen in Figure 5. The original purpose of these demo applications was to facilitate a qualitative review of our model's performance. I had arranged for this with a radiology superintendent from Kent and Medway Hospital; however, due to his limited availability, it was not possible to proceed as planned. Despite this setback, the demo application proved to be extremely useful for testing localisation methods in this project and became an invaluable resource for both development and testing.

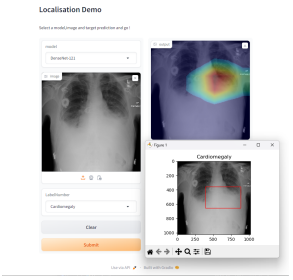

Fig. 4. Screenshot of demo application for localisation. The bottom right is a radiologist annotated Chest X-ray with cardiomegaly, and the application is showing a DenseNet-121 models localisation on this image

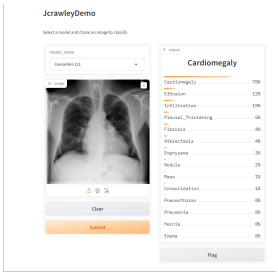

Fig. 5. Screenshot of demo application for classification. The model loaded is DenseNet-121 with an image of an X-ray which has ground truth as Cardiomegaly

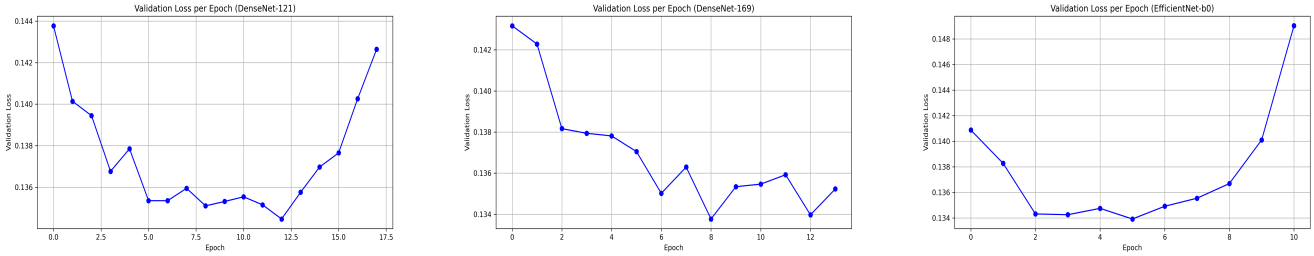

Fig. 6. Validation loss per Epoch for: DenseNet-121 (left),DenseNet-169 (middle) and EfficientNet-b0 (right)

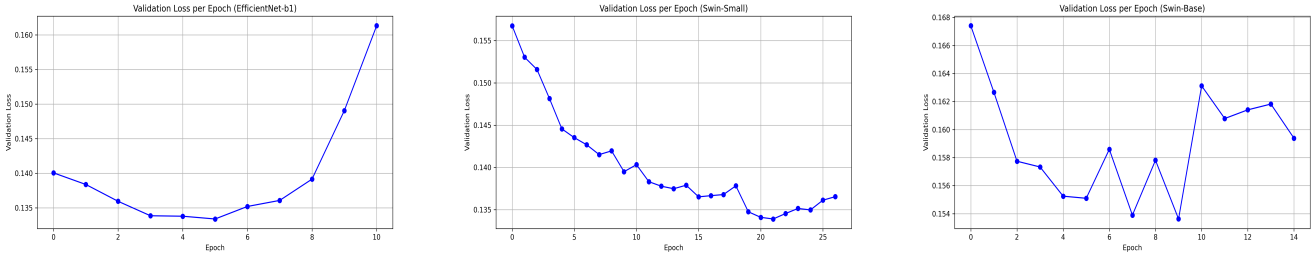

Fig. 7. Validation loss per Epoch for EfficientNet-b1 (left), Swin-Small (middle) and Swin-Base (right)

TABLE 5
AUCs of Different Models

Model Name	Atel	Cons	Infi	Pneu1	Edem	Emph	Fibr	Effu	Pneu2	Pleu	Card	Nodu	Hern	Mass	Macro Average
ENet-b0	0.76	**0.75**	0.69	0.85	**0.84**	**0.90**	0.80	0.82	0.71	0.76	0.87	**0.76**	0.87	**0.81**	0.8007
ENet-b1	**0.77**	0.74	**0.70**	**0.86**	**0.84**	**0.90**	**0.82**	0.82	0.71	**0.78**	0.87	0.74	0.86	**0.81**	0.8017
DenseNet-121	0.76	**0.75**	**0.70**	0.84	**0.84**	0.88	**0.82**	0.82	**0.72**	0.77	**0.89**	0.74	0.89	**0.81**	0.8021
DenseNet-169	0.76	0.74	0.69	0.84	**0.84**	**0.90**	0.81	**0.83**	0.71	0.76	**0.89**	0.74	**0.91**	**0.81**	**0.8024**
Swin-Small	0.75	0.74	**0.70**	0.84	0.83	0.88	0.81	0.82	**0.72**	0.76	0.88	0.74	0.88	0.80	0.7966
Swin-Base	0.67	0.69	0.65	0.71	0.76	0.66	0.72	0.74	0.60	0.65	0.68	0.62	0.78	0.59	0.6806

TABLE 6
Comparison of AUCs with literature and our best model

Model Name	Atel	Cons	Infi	Pneu1	Edem	Emph	Fibr	Effu	Pneu2	Pleu	Card	Nodu	Hern	Mass	Average
DenseNet-169 (ours)	0.76	0.74	0.69	0.84	0.84	0.90	0.81	0.83	0.71	0.76	**0.89**	0.74	0.91	0.81	0.8024
Yan et al. [36] (SOTA)	**0.792**	**0.7598**	**0.7095**	**0.8759**	0.8478	**0.9422**	**0.8326**	**0.8415**	**0.7397**	**0.8083**	0.8814	**0.8105**	**0.9341**	**0.8470**	**0.8302**
Gundel et al. [10]	0.767	0.745	0.709	0.846	0.835	0.895	0.818	0.828	0.731	0.761	0.883	0.758	0.896	0.821	0.807
Taslimi et al. [33]	0.781	0.748	0.701	0.871	**0.848**	0.914	0.826	0.824	0.713	0.778	0.875	0.78	0.855	0.822	0.81

TABLE 7
IouS at various thresholds of our models

Model Name - Threshold	Atel	Card	Effu	Infi	Mass	Nodu	Pneumonia	Pneumothorax	Avg
DenseNet-121 - 0.1	0.06	0.34	0.1	0.16	0.06	0.01	0.14	0.07	0.117
DenseNet-121 - 0.15	0.06	0.35	0.1	0.16	0.08	0.01	0.15	0.07	0.121
DenseNet-121 - 0.25	0.06	0.371	0.1	0.174	0.08	0.01	0.17	0.07	0.129
DenseNet-121 - 0.35	0.07	0.39	0.12	0.18	0.07	0.01	0.18	0.06	0.133
DenseNet-121 - 0.45	0.07	0.32	0.11	0.19	0.08	0.01	0.15	0.05	0.123
DenseNet-169 - 0.1	0.06	0.33	0.09	0.135	0.07	0.01	0.12	0.03	0.116
DenseNet-169 - 0.15	0.06	0.37	0.09	0.17	0.06	0.01	0.14	0.05	0.119
DenseNet-169 - 0.25	0.07	0.40	0.12	0.14	0.08	0.01	0.16	0.04	0.127
DenseNet-169 - 0.35	0.08	0.394	0.11	0.14	0.09	0.01	0.16	0.05	0.127
DenseNet-169 - 0.45	0.06	0.33	0.09	0.13	0.07	0.01	0.12	0.03	0.105
ENet-b0 - 0.1	0.03	0.150	0.08	0.11	0.03	0.01	0.09	0.02	0.066
ENet-b0 - 0.15	0.03	0.14	0.075	0.11	0.04	0.005	0.01	0.03	0.066
ENet-b0 - 0.25	0.04	0.13	0.08	0.11	0.03	0.01	0.1	0.02	0.065
ENet-b0 - 0.35	0.04	0.11	0.09	0.11	0.05	0.01	0.1	0.02	0.065
ENet-b0 - 0.45	0.04	0.1	0.09	0.11	0.05	0.001	0.12	0.015	0.067
ENet-b1 - 0.1	0.05	0.26	0.09	0.14	0.06	0.01	0.134	0.07	0.102
ENet-b1 - 0.15	0.05	0.29	0.1	0.15	0.07	0.01	0.14	0.07	0.108
Enet-b1 - 0.25	0.06	0.33	0.11	0.16	0.08	0.01	0.16	0.06	0.121
ENet-b1 - 0.35	0.06	0.35	0.11	0.17	0.09	0.01	0.18	0.07	0.130
ENet-b1 - 0.45	0.07	0.31	0.13	0.18	0.08	0.01	0.18	0.06	0.128
Swin-Small - 0.1	0.05	0.24	0.07	0.11	0.04	0.01	0.1	0.03	0.081
Swin-Small - 0.15	0.06	0.26	0.08	0.10	0.03	0.01	0.09	0.03	0.081
Swin-Small - 0.25	0.06	0.26	0.06	0.10	0.03	0.01	0.09	0.03	0.077
Swin-Small - 0.35	0.05	0.24	0.05	0.07	0.03	0.01	0.07	0.02	0.067
Swin-Small - 0.45	0.05	0.20	0.04	0.06	0.02	0.01	0.07	0.02	0.057
Swin-Base - 0.1	0.03	0.13	0.06	0.08	0.03	0.01	0.08	0.04	0.056
Swin-Base - 0.15	0.02	0.13	0.06	0.07	0.03	0.01	0.07	0.04	0.054
Swin-Base - 0.25	0.02	0.10	0.05	0.07	0.02	0.01	0.07	0.04	0.046
Swin-Base - 0.35	0.02	0.08	0.04	0.06	0.02	0.001	0.07	0.03	0.039
Swin-Base - 0.45	0.01	0.06	0.04	0.05	0.02	0.01	0.05	0.03	0.032

TABLE 8
IouS at of EigenCam visualisation of our models

Model Name - Threshold	Atel	Card	Effu	Infi	Mass	Nodu	Pneumonia	Pneumothorax	Avg
DenseNet-121 - 0.1	0.07	0.413	0.123	0.19	0.07	0.01	0.17	0.06	0.137
DenseNet-169 - 0.1	0.07	0.42	0.11	0.18	0.08	0.01	0.14	0.05	0.135
EfficientNet-b0 - 0.1	0.07	0.32	0.11	0.2	0.09	0.01	0.18	0.08	0.133
EfficientNet-b1 - 0.1	0.04	0.21	0.08	0.11	0.054	0.01	0.113	0.06	0.08
Swin-Small - 0.1	0.05	0.25	0.08	0.12	0.05	0.01	0.11	0.04	0.09
Swin-Base - 0.1	0.03	0.11	0.05	0.04	0.01	0.002	0.05	0.02	0.04

5 Conclusion

In this project we trained 6 deep-learning image classifiers to the task of multi-label Chest X-ray classification via transfer learning. We used the NIH Chest-x-ray-14 [34] dataset to facilitate model training and evaluation. Following the training of these models, we evaluated them on the official test split, obtaining an AUC for each class. The classification AUC for each model was impressive, at about 0.8, with the exception of our Swin-Base model. Our DenseNet-169 model performed the best out of our models. We then compared our models with those studied in the literature and found our DenseNet-169 model yielded comparable results to the state-of-the-art model in a number of classes and even yielded an improvement in the 'cardiomegaly' class.

We conducted a localisation study on each of our developed models using a range of thresholds, as indicated by their effectiveness in the literature. This study yielded suboptimal results due to our unconventional methods of model localisation; specifically, we did not generate bounding boxes but instead used heatmaps. Despite this deviation from standard practices, our results were comparable to other models within this project. Notably, our DenseNet models, while achieving similar classification performance, exhibited a significant improvement in localisation compared to both the EfficientNet and Swin models.

After training and evaluation of classification was completed, we developed demo applications to showcase our model's localisations and classifications.

We have successfully addressed our Research question: *Which state-of-the-art deep learning classification models yield the best classification and localisation performance in transfer-learning for multi-label Chest X-ray image classification ?* with our DenseNet-169 model, which achieved the highest classification and localisation performance in our studies. Additionally, our provided localisation study showcased the localisation performance of recent state-of-the-art classifiers (Swin Transformers), a gap identified within the literature, as the localisation of these models is yet to be evaluated on the task of CXR classification.

5.1 Limitations and Future work

Our project achieved its objectives and effectively addressed the research question. Nevertheless, there remains considerable potential for further enhancements and development. Below, we outline potential avenues for future work and discuss some limitations of the current project.

- Exploring hyperparameter optimisation techniques, such as Grid Search, to maximize training efficiency for each model.
- Incorporating bounding box data into a training algorithm, using a loss function that trains models for localisation.
- Further the quantitative evaluation on localisation performance by using different visualisation techniques such as Grad-CAM++ and Deep Feature Factorisation.
- Utilise multiple datasets to perform a cross-dataset evaluation on classification and localisation.
- Conducting a localisation study using bounding boxes generated by model heatmaps.

6 Acknowledgement

The author would like to thank Dr. Stamos Katsigiannis for his invaluable support given throughout this project.

References

[1] Abubakar Abid, Ali Abdalla, Ali Abid, Dawood Khan, Abdulrahman Alfozan, and James Y. Zou. Gradio: Hassle-free sharing and testing of ML models in the wild. *CoRR*, abs/1906.02569, 2019.

[2] Saif Sarmad Al-Shamari and Hilal Al-Libawy. An accurate deep learning threat image detection algorithm for x-ray baggage dataset. In *2022 International Conference for Natural and Applied Sciences (ICNAS)*, pages 5–10, 2022.

[3] Ivo M. Baltruschat, Hannes Nickisch, Michael Grass, Tobias Knopp, and Axel Saalbach. Comparison of deep learning approaches for multi-label chest x-ray classification, 2019.

[4] H. C. Becker, W. J. Nettleton, P. H. Meyers, J. W. Sweeney, and C. M. Nice. Digital computer determination of a medical diagnostic index directly from chest x-ray images. *IEEE Transactions on Biomedical Engineering*, BME-11(3):67–72, 1964.

[5] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Sorelle A. Friedler and Christo Wilson, editors, *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, volume 81 of *Proceedings of Machine Learning Research*, pages 77–91. PMLR, 23–24 Feb 2018.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[9] Andre Esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, Feb 2017.

[10] Sebastian Guendel, Sasa Grbic, Bogdan Georgescu, Siqi Liu, Andreas Maier, and Dorin Comaniciu. Learning to recognize abnormalities in chest x-rays with location-aware dense networks. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 23rd Iberoamerican Congress, CIARP 2018, Madrid, Spain, November 19-22, 2018, Proceedings 23*, pages 757–765. Springer, 2019.

[11] Ameer Hamza, Muhammad Attique Khan, Shui-Hua Wang, Abdullah Alqahtani, Shtwai Alsubai, Adel Binbusayyis, Hany S. Hussein, Thomas Markus Martinetz, and Hammam Alshazly. Covid-19 classification using chest x-ray images: A framework of cnn-lstm and improved max value moth flame optimization. *Frontiers in Public Health*, 10, 2022.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks, 2016.

[14] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks, 2018.

[15] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

[16] Alistair EW Johnson, Tom J Pollard, Nathaniel R Greenbaum, Matthew P Lungren, Chih-ying Deng, Yifan Peng, Zhiyong Lu, Roger G Mark, Seth J Berkowitz, and Steven Horng. Mimic-cxr-jpg, a large publicly available database of labeled chest radiographs. *arXiv preprint arXiv:1901.07042*, 2019.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[18] Jakub Kufel, Michał Bielówka, Marcin Rojek, Adam Mitrega, Piotr Lewandowski, Maciej Cebula, Dariusz Krawczyk, Marta Bielówka, Dominika Kondoł, Katarzyna Bargieł-Łaczek, Iga Paszkiewicz, Łukasz Czogalik, Dominika Kaczyńska, Aleksandra Wocław, Katarzyna Gruszczyńska, and Zbigniew Nawrat. Multi-label classification of chest x-ray abnormalities using transfer learning techniques. *Journal of Personalized Medicine*, 13(10), 2023.

[19] Nataliia Kussul, Mykola Lavreniuk, Sergii Skakun, and Andrii Shelestov. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, 14(5):778–782, 2017.

[20] Zhe Li, Chong Wang, Mei Han, Yuan Xue, Wei Wei, Li-Jia Li, and Li Fei-Fei. Thoracic disease identification and localization with limited supervision, 2018.

[21] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021.

[22] Mohammed Bany Muhammad and Mohammed Yeasin. Eigen-cam: Class activation map using principal components. In *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2020.

[23] Assosiation of American Medical Colleges. Aamc report reinforces mounting physician shortage. 2021.

[24] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks, 2015.

[25] Aravind Sasidharan Pillai. Multi-label chest x-ray classification via deep learning. *Journal of Intelligent Learning Systems and Applications*, 14(04):43–56, 2022.

[26] Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu, Brandon Yang, Hershel Mehta, Tony Duan, Daisy Yi Ding, Aarti Bagul, Curtis P. Langlotz, Katie S. Shpanskaya, Matthew P. Lungren, and Andrew Y. Ng. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. *CoRR*, abs/1711.05225, 2017.

[27] Eyal Rozenberg, Daniel Freedman, and Alex Bronstein. Localization with limited annotation for chest x-rays, 2019.

[28] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[29] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, October 2019.

[30] Joe Spizzandre. swin vs vit :, Feb 2024.

[31] Hossam H. Sultan, Nancy M. Salem, and Walid Al-Atabany. Multi-classification of brain tumor images using deep neural network. *IEEE Access*, 7:69215–69225, 2019.

[32] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks, 2020.

[33] Sina Taslimi, Soroush Taslimi, Nima Fathi, Mohammadreza Salehi, and Mohammad Hossein Rohban. Swinchex: Multi-label classification on chest x-ray images with transformers, 2022.

[34] Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald Summers. Chestx-ray14: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. 09 2017.

[35] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

[36] Chaochao Yan, Jiawen Yao, Ruoyu Li, Zheng Xu, and Junzhou Huang. Weakly supervised deep learning for thoracic disease classification and localization on chest x-rays. In *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, BCB '18. ACM, August 2018.

[37] Huanhuan Zhang and Yufei Qie. Applying deep learning to medical imaging: A review. *Applied Sciences*, 13(18), 2023.

[38] Claire S. Zhu, Paul F. Pinsky, Barnett S. Kramer, Philip C. Prorok, Mark P. Purdue, Christine D. Berg, and John K. Gohagan. The Prostate, Lung, Colorectal, and Ovarian Cancer Screening Trial and Its Associated Research Resource. *JNCI: Journal of the National Cancer Institute*, 105(22):1684–1693, 10 2013.

[39] . Reminder of the first paper on transfer learning in neural networks, 1976. *Informatica*, 44, 09 2020.