DEEP LEARNING MODELS FOR CLASSIFYING FALSE POSITIVES IN SEGMENTED OSTEOLYTIC BONE LESION CANDIDATES FROM CT SCANS

MUNIRDIN JADIKAR

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
AT THE SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
OF TILBURG UNIVERSITY

STUDENT NUMBER

2079517

COMMITTEE

dr. L.L. Sharon Ong
ir. Martijn van Leeuwen MSc
dr. Javad Pourmostafa Roshan Sharami

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

January 13, 2023

WORD COUNT: 8350

ACKNOWLEDGMENTS

It is an honor to be a part of the WeCare project team, with excellent professors and biomedical engineers. I'd like to express my gratitude to my supervisor, Dr. Lee-Ling Sharon Ong, for her guidance and encouragement throughout this thesis. I want to thank ir. Martijn van Leeuwen, M.Sc., for his involvement and great technical suggestions for this thesis. My appreciation also goes to my wife and two daughters for their encouragement and support during my study.

DEEP LEARNING MODELS FOR CLASSIFYING FALSE POSITIVES IN SEGMENTED OSTEOLYTIC BONE LESION CANDIDATES FROM CT SCANS

MUNIRDIN JADIKAR

Abstract

This thesis aims to build a lesion classification model to classify False Positives(FPs) from segmented bone lesion (tumor) candidates. Without a radiologist's expertise, the FPs from segmented bone lesions are hard to differentiate from True Positives(TPs). To address this issue, a lesion classification model was developed using Deep Convolutional Neural Network (DCNN) models. The study utilizes transfer learning (TL), which is an effective method for medical image classification. The research question addressed in this thesis is "To what extent can DCNN models classify potential FPs or missed lesion candidates in segmented osteolytic bone lesions?" The datasets used for this study were extracted from CT scans of patients with multiple myeloma from Elizabeth-Tweesteden (ETZ) hospital. Due to the limited data, different pre-trained DCNN models were explored. This study's results showed that the fine-tuned EfficientNetB7 model achieved a test accuracy of 0.93 and an F1 score of 0.93 on the test set. Additionally, some FPs from the bone segmentation model was labeled by expert radiologists with correct classes, resulting in a hold-out test set. The best-performing model, EfficientNetB7, achieved an F1 score of 0.69 and a False Positive Rate (FPR) of 0.42 on the hold-out test set. The ensemble learning method applied to the fine-tuned models was found to be effective in classifying and reducing FPs. By incorporating a classifier into the segmentation process, the number of FPs can be reduced, leading to a more reliable system and a reduced workload for radiologists.

1 ETHICAL STATEMENT

Work on this thesis did not involve collecting data from human participants or animals. The dataset used in this study belongs to Elisabeth-Tweesteden hospital, and the author acknowledges that it does not have any legal claim to this data. The medical images are generated from the dataset of the hospital, and the author did not have any legal claim on these medical images.

2 INTRODUCTION

Multiple myeloma (MM) is a plasma cell cancer. It can lead to, for instance, bone lesions (tumors), anemia, and other severe symptoms (Filho et al., 2019). Between 80% to 90% of MM patients develop osteolytic bone lesions during the course of their disease (Rajkumar & Kumar, 2016). Therefore, detecting lesions is crucial for diagnoses and follow-ups. One of the imaging methods to examine lesions is Computed Tomography (CT) imaging of the whole body (Xu et al., 2018). CT scans provide radiologists with valuable information to visually detect and measure bone lesions, which appear as small holes in a CT scan, giving the bone a "punched-out" look (Reagan, Liaw, Rosen, & Ghobrial, 2015). Furthermore, osteolytic bone lesions are small and can be found in multiple locations in the body. As a result, the daily error rate of a radiologist lies between 3-5%. However, in cross-sectional imaging, the interpretation errors lie between 20%-30% (Maskell & Frcp, 2019). In addition, radiologists must possess the expertise to detect bone lesions (Y. He et al., 2020). Because of these reasons, there is a possibility that radiologists overlook certain lesions when evaluating a patient's CT scan.

In recent years, there has been a significant increase in attention toward the application of deep learning in medical imaging. A variety of breakthroughs have been achieved through the utilization of deep learning algorithms for the classification and segmentation of different types of lesions. One such algorithm, the U-Net, has proven to be particularly effective in the segmentation of bone lesions (Ronneberger, Fischer, & Brox, 2015). The We Care project team applied a 2D U-net to the bone lesions. However, it should be noted that when utilizing a 2D U-net to segment bone lesions, the model may segment certain regions as lesions that were not labeled as such in the ground truth. These regions may be FPs or lesions that were missed by radiologists, thus making it challenging to determine whether they are TPs or true negatives (TNs).

By classifying the output of the segmentation results as lesions or non-lesions, it is possible to determine whether an FP from a segmented

bone lesion belongs to the lesion or non-lesion class. Figure 1 shows some examples of FPs predictions produced by the 2D U-net. The highlighted regions are FPs, and without radiologists' expertise, it is hard to differentiate whether these FPs belong to lesion and non-lesion classes. Thus, developing a classifier based on deep learning will improve the segmentation framework's reliability and decrease the workload of radiologists.

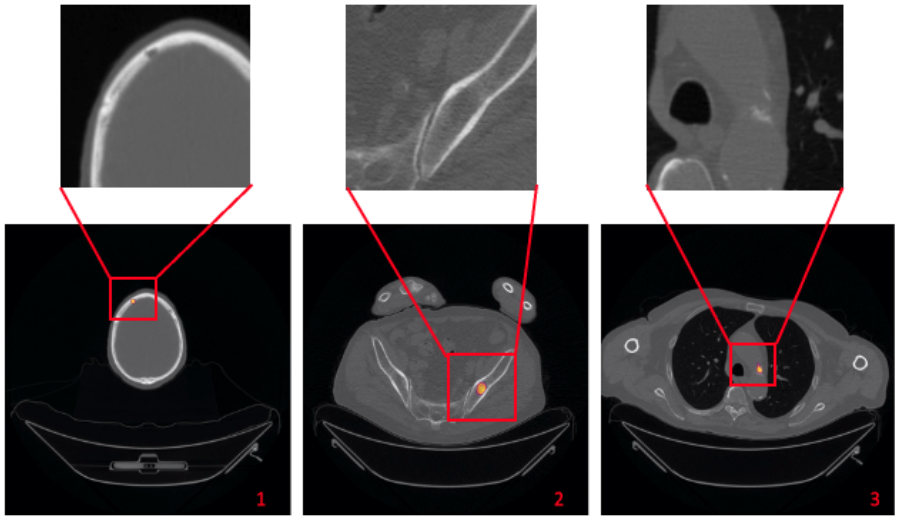


Figure 1: FPs in segmented bone lesions. 1: Granulation, 2: Fat tissue, 3: Aorta.

2.1 *Motivation*

MM is often diagnosed at later stages due to its rarity and the similarity of symptoms with other types of diseases (Koshiaris, 2019). For example, MM patients are often misdiagnosed as having rheumatoid arthritis (Schoninger, Homsi, Kreps, & Milojkvovic, 2018). However, early-stage diagnosis can significantly increase patients' survival rates and positively impact patient treatment. Thus, a classification model can aid radiologists in classifying bone lesions in a timely manner. The developed classification model will also be utilized to evaluate the FPs from the segmentation algorithm, thereby accelerating the deployment of the segmentation framework. Furthermore, the classification model could be integrated into the processing of CT scans to predict beforehand whether a patient has a bone lesion or not.

From a scientific perspective, this study represents the first investigation of using a classification model in the post-processing of the segmentation model to classify FPs. As such, it is not only relevant for classifying bone lesions but can also be applied in other medical fields. Additionally, the

techniques employed in this study, such as TL and ensemble learning with limited data, can be applied to other medical classification tasks. Ultimately, the findings of this study will contribute to our understanding of how to effectively handle FPs in segmented bone lesions.

2.2 *Research questions*

DCNN models have achieved state-of-the-art performance in image classification and have shown exceptional classification performances for certain types of cancer. However, there is a lack of research on utilizing DCNN models for the classification of FPs from segmented bone lesions. This study aims to address this gap by applying pre-trained DCNN models to classify FPs in segmented bone lesions. The main research question for this thesis is:

> *To what extent can DCNN models classify potential PFs or missed lesion candidates in segmented osteolytic bone lesions?*

TL has gained significant popularity for datasets with limited size, and pre-trained models with ImageNet weights have exhibited remarkable results on limited medical images (Deepak & Ameer, 2019). Additionally, it has been demonstrated that pre-trained models can effectively extract image features from medical images and classify various types of diseases (Litjens et al., 2017). Thus, this thesis will focus on identifying and applying pre-trained models for bone lesion classification. Consequently, the first subquestion in this study is:

RQ1 *To what extent can TL increase classification model performance?*

The radiologists have reviewed some FPs from the segmented bone lesions, resulting in a small hold-out test set. These FPs have been re-labeled with their corresponding actual classes. Thus, it is crucial to investigate the performance of the final model on this hold-out test set. Therefore, the next subresearch question in this study will be:

RQ2 *To what extent can DCNN models generalize to the dataset created from the feedback of radiologists ?*

Combining the predictions of models could increase the performance of the classification. For example, one model could be good at classifying FPs, while the other is good at classifying FNs. Combining the predictions of the models or applying other types of ensemble learning techniques could increase the performance of the classification task. Therefore, the following subreserach question is:

RQ3 *To what extent can ensemble learning improve the classification result?*

This thesis is part of the WeCare project, a collaboration between Tilburg University and ETZ hospital, so the ETZ hospital provides the CT scans. Patches for model training and testing are extracted from these CT scans. The baseline model, VGG16, was trained on three different image resolutions and performed well on a resolution of 224x224. Subsequently, VGG16, InceptionV3, ResNet50, and EfficienNetB7 were selected based on a comprehensive literature review. These pre-trained models were implemented with ImageNet weights and subsequently fine-tuned for optimal performance. The final results indicate that the fine-tuned EfficientNetB7 model achieved the highest performance. This model was further tested on the test set and hold-out test set. The test set results indicated that EfficientNetB7 achieved an F1-score of 0.93. The generalizability test results demonstrated that EfficientNetB7 reached an F1 score of 0.69 on the hold-out test set. Finally, an ensemble learning method was applied, and the final result indicated that stacking the predictions of four models resulted in an F1-score of 0.85 and a FPR of 0.15.

3 RELATED WORK

3.1 *Related work on FP reduction on classification and segmentation*

Z. Yang et al. (2022) proposed using a deep-learning and ensemble classifier to reduce PFs from brain metastases (BMs) segmentation. This classifier consisted of a Siamese network and a support vector machine (SVM) classifier, and it was designed to reduce the FPR in segmentation. When tested on a dataset of segmented multiple BMs, the classifier achieved accuracy, sensitivity, specificity, and area under the curve of 0.91, 0.96, 0.90, and 0.93, respectively. When the model was integrated into the original segmentation platform, it significantly reduced the FPR, making it a helpful tool for BMs segmentation.

Zhao, Liu, Yin, and Wang (2022) presented a new method for reducing FPs in automated pulmonary nodule detection using multi-scale CNNs. The existing techniques for FPs reduction mainly use 3D CNNs, but these methods have long training times and require frequent retraining to maintain optimal performance. The proposed method addresses these issues using three different 2D images cropped from 3D CT scans to preserve spatial information and shorten training time. The results showed that the proposed method decreased training time from 36 hours to 8 hours and achieved a sensitivity of 0.95 and specificity of 0.98.

Another paper proposed a novel computer-aided detection (CAD) model for pulmonary nodules using multi-view CNN (Setio et al., 2016). The model was designed to learn discriminative features from training data automatically. The nodules can be classified as solid, subsolid, and large, and a set of 2D patches from different orientations is extracted for each class. The proposed architecture consists of multiple streams of 2D ConvNets, and the outputs are combined using a reliable fusion method to get the final classification. This method was tested on a publicly available LIDC-IDRI dataset and achieved high detection sensitivities of 0.85 and 0.90 at 1 and 4 FPs per scan, respectively.

3.2 *Related works on bone lesion classification*

Deep learning algorithms have previously been applied to bone lesions, for example, to classify lesions in MRI scans. Eweje et al. (2021) proposed an ensemble deep learning which consists of EfficientNetBo and logistic regression. The ensemble model consists of EfficientNet trained on T1-Weighted and T2-Weighted images and a logistic regression model based on clinical features. This model was evaluated on a test set alongside the result of three experienced radiologists. Then, different performance measurements were used for evaluation, including F2-score, accuracy, sensitivity, and specificity. The outcome suggested that the ensemble model performs at near expert level.

A deep learning model was proposed to bone tumors into three classes (Y. He et al., 2020); benign, intermediate, and malignant. The model's performance was assessed using external datasets and compared with the classification results of five leading radiologists. The interesting point in this study was using 3-channeled images for the deep learning model, which were created based on two-dimensional slices of CT scans. The deep learning model for this research was EffcientNEt-BO, which had pre-trained weights from the ImageNet database. The model predictions were compared with radiologists' results. The multi-labels classification model (3 classes) achieved the same level of grouping accuracy as specialists. In addition, the multi-class model performed better than junior radiologists with 6 and 7 years of experience.

Another study proposed to extract training data patches by applying different approaches (Perkonigg, Hofmanninger, Menze, Weber, & Langs, 2018). First, the positive class patches were extracted from the 3D CT scans according to the center of the bone lesion with data augmentation techniques like mirroring and random rotation. Next, the negative class patches were extracted from CT scans based on random positions in a bone mask without positive classes. For the second approach, 3-channeled

patches were extracted using three ranges of Hounsfield Units (HU) for each channel. Frequently, the first channel has less than 100 HU, the second channel is higher than 400 HU, and the last is between 100 and 400 HU. All images for single and multi-channel are reshaped to 64×64. The final result showed that a pr-trained model with 3-channeled patches performs better with fewer FPs.

3.3 *Related works on TL for medical imaging data*

Computer-based lesion detection methods lack large scales of annotated datasets. To solve this problem, Yan, Wang, Lu, and Summers (2018) collected a large-scale radiological imaging dataset with annotations based on the RECIST[1] bookmarks on CT scans. The VGG16 model architecture was adopted as a base model, and it was compared to pre-trained DenseNEt120, ALexNet, and ResNet50 on a separate validation set. The final result indicated that the VGG16 model performed differently for each type of lesion. The liver and longs lesions were classified with higher sensitivity, and it was due to the intensity and appearance of the lesion on the image. Also, these lesions made up a large part of the training data. Noticeably, the bone lesions had the lowest sensitivity in the testing dataset. This is because of the lesions' contrast and sizes in the patches.

Litjens et al. (2017) reviewed 102 studies on TL for medical image analysis. The authors found that AlexNet was mainly used for brain-related tasks. Besides, the Inception model was applied for skeletal systems (57%), and the VGGNet was mainly used for eye-related diseases (42%). This review provided a comprehensive overview of the TL techniques which could be applied in this thesis.

Not all papers agreed on the improvements of TL in medical image analysis. Raghu, Zhang, Kleinberg, and Bengio (2019) suggested that TL cannot constantly improve the model performance on limited medical data since the ImageNet architectures were over-parameterized. Therefore, the authors compared simple CNN architectures with deep ResNet and InceptionV3 models. All models were trained using pre-trained and randomly initialized weights on the dataset. Interestingly, the performance of the simple CNN models with random weight initialization was comparable to those of the ResNet50 and InvceptionV3 models on a large dataset. On the other hand, using larger models like ResNet50 with pre-trained weights for a small dataset did improve the performance by 2%.

[1] RECIST is a common method to measure how well a cancer patient is responding to therapy.

3.4 *Related work on using an ensemble of classifiers*

One of the reasons why ensemble-based methods are so widely used in data stream classification is because they tend to perform well compared to other single solid learners, and they are relatively straightforward to implement in real-world scenarios. Müller, Soto-Rey, and Kramer (2022) examined ensemble learning techniques for medical image classification, including Augmenting, Stacking, and Bagging. Stacking was found to have the most significant performance gain while Augmenting consistently improved non-overfitting models, and it was applied to single-model pipelines. Bagging demonstrated significant performance gain similar to Stacking but relied on sampling with sufficient feature representation in all folds. Simple pooling functions like Mean or Majority Voting were often as effective or better than more complex pooling functions like Support Vector Machines. Another study proposed a stacking ensemble deep learning model based on 1D-CNNs for multi-class classification of five common types of cancer using genetic data((Mohammed, Mwambi, Mboya, Elbashir, & Omolo, 2021). The model was compared to a single 1D-CNN and several machine learning methods, including support vector machines. The results showed that the proposed model performed better than the other classifiers. Due to the limited amount of labeled medical data, medical image classification could suffer from training problems such as overfitting, local optimums, and vanishing gradients. Y. Yang, Hu, Zhang, and Wang (2022) proposed a two-stage selective ensemble approach using deep tree training (DTT). This strategy involved jointly training a series of networks hierarchically constructed from the hidden layers of CNNs, resulting in better performance.

3.5 *Knowledge gap*

There is a scarcity of literature about reducing FPs in bone lesion classification. At the same time, no studies have been done about classifying FPs from segmented bone lesions. Additionally, only a few studies have focused on enhancing the quantity and quality of medical datasets. While TL on medical images has been extensively studied, there is limited research on the classification of bone lesions. Previous studies have demonstrated the effectiveness of ensemble learning methods; however, these have not been utilized to classify FPs arising from bone lesion segmentation. Therefore, the primary objective of this study is to investigate the application of pre-trained DCNN models with ensemble learning methods for the classification of FPs from segmented bone lesions.

4 METHODOLOGY AND EXPERIMENTAL SETUP

4.1 *Dataset Description*

The dataset used in this thesis belongs to The ETZ hospital in Tilburg. The dataset is anonymized for all demographic information, and it consists of 96 CT scans from 79 patients diagnosed with MM. The original CT scans are in Digital Imaging and Communications in Medicine (DICOM) format. These images are converted to NIFTI format, a common format in medical imaging. The shape of each axial slice is either 512×512 pixels or 768×768 pixels, and each slice has a thickness of 2.5 mm or 3 mm. The lesions were annotated by expert radiologists from ETZ hospital using 3D Slicer and converted into NIFTI file format. In addition, the bones in each scan were segmented using a U-Net algorithm (Tong, 2021), resulting in bone-map data. Therefore, for this thesis, CT scans, label data, and bone maps are used to generate patches.

4.2 *Data preprocessing*

4.2.1 *Patch extraction*

The CT scans are in 3D NIFTI format, but DCCN models with 2D convolutional layers require 2D images; therefore, 2D patches should be extracted with and without bone lesions. The ground truth labels are in 3D NIFTI format. Each slice of the CT scan can be indexed according to the slice indices (on the Z-axis). The annotated lesions from CT scans can be cropped around the center of the lesion. Then, the location of the lesions is obtained by applying connected components to group pixels into regions. Scikit-image's regionprops module returns the properties of labeled regions, such as size, center, and perimeter (Gouillart, Nunez-Iglesias, & Walt, 2017). The CT scans and labels are iterated according to the slice indices, and the regionprops function is run on the slice that contains the label. The center of each labeled lesion is gathered from the centroids. Finally, the CT-scan slice is cropped around the lesion's center.

Extracting a lesion-free patch at a random location on a slice of a CT scan could result in a patch without any bone pixels because the proportion of the bone in a slice of a CT scan is relatively small. One way to get the location of bone is by running a 2D-Unet segmentation algorithm trained on the CT-ORG dataset to segment bones in a CT-scan (Rister, Yi, Shivakumar, Nobashi, & Rubin, 2020). When the segmented bone map is created in NIFTI binary format, it is possible to extract lesion-free patches from the CT scans based on the bone map. Each CT-scan slice is cropped around

a randomly selected bone pixel from the cropped bone map slice, and an extra condition was added to exclude the bone lesions from these patches. Since the no-lesion patches are cropped around a bone pixel, the patch might have significantly small bone pixels on a patch; therefore, patches are obtained with a condition of a minimum of 10% bone pixels in every non-lesion patch.

4.2.2 *Image resolution*

One of the crucial parameters for DCNN models is image resolution. Deep learning algorithms require large image sizes because DCNN models can extract more features from larger images. However, most pre-trained DCNN models are trained on an image resolution between 200×200 and 300×300. Increasing the input size beyond 224 x224, these models have a plateau effect on model performance (Sabottke & Spieler, 2020). Each pre-trained model is trained on a specific image resolution. When the top layers of the pre-trained model are not included, it is possible to use different input image resolutions. However, the model's performance might differ on each image resolution. Therefore, this study will experiment with different image resolutions on the baseline model, so the patches are extracted in 100×100,192×192, and 224×224 resolutions. The InceptiopV3 requires an input size of between 75×75 to 299×299 pixels; therefore, 100×100 is a reasonable safe input size for the selected pre-trained models. The 192×192 is the input shape of the U-Net segmentation algorithm; however, it is interesting to test this resolution on classification algorithms. Besides, the 224×224 image resolution is chosen because it is a common input shape for most pre-trained DCNN models. The baseline model will be trained on these three image resolutions, and the results will be compared to determine the best one.

4.2.3 *Final datasets*

Table 1 shows the number of samples in each dataset. The extracted patches are splited into training, validation, and test set. The hold-out test set contains a few segmented bone lesions which are previously classified as FPs by the U-Net and reviewed by a radiologist with correct labels. However, the hold-out test set is imbalanced.

	Lesion	Non-lesion
Training data	2200	2235
Validation data	550	560
Test data	356	355
Hold-out test data	18	227

Table 1: The number of samples in dataset

4.3 *Pre-trained models*

TL is popular in the medical field with limited labeled data. Most publicly available medical datasets are not adequately labeled, and the variety of diseases makes it more challenging to find a dataset set with appropriate labels. As a result, TL is preferred by medical AI researchers. Most pre-trained models are trained on the ImageNet dataset, so the models contain weights of natural images; even so, the pre-trained models performed well on medical images (Hossain, Iqbal, Islam, Akhtar, & Sarker, 2022). This section will briefly explain four different pre-trained models for this thesis.

4.3.1 *VGG16*

VGG16 has a shallow structure than other pre-trained models (Simonyan & Zisserman, 2014). VGG16 is mainly applied as the baseline model in medical AI because it requires less computational power and time than any other pre-trained models (Yan et al., 2018). Moreover, despite its shallow architecture, it can reach high accuracy for TL tasks. This model consists of 13 convolutional layers with 3×3 filters (Figure 2). The convolutional layers can be divided into five blocks, and a max pooling layer follows each. The final max pooling layer is connected to a flatten layer followed by three dense layers. The final dense layers have 1000 units with a softmax activation function. TL tasks without a modified VGG16 model require an input image shape of 224×224 (Simonyan & Zisserman, 2014). When its final dense layers are removed, it can be modified with different dense layers with different input image shapes. For this thesis, the VGG16 model is chosen as the baseline model because it is very convenient and easy to apply for TL tasks.

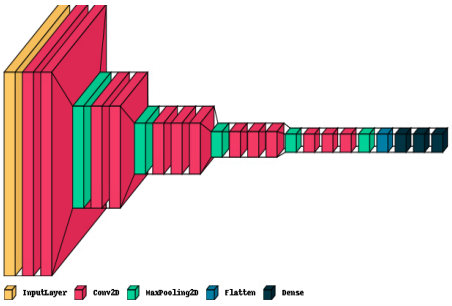


Figure 2: VGG16 model structure. Generated from keras-visualizer

4.3.2 *InceptionV3*

The inceptionV3 architecture contains multiple inception modules (Figure 3), and they are stacked upon each other (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2015). The main benefit of this type of module is reducing the computation and overfitting. The key to improvement in InceptionV3 comes from the new way of computing the convolutional operation. In the previous version of the Inception model, inception modules had 5×5 filters, replaced by 3×3 filters in InceptionV3. This ensured that it had fewer parameters for training, so it needed less computational power and time. Another interesting fact is applying asymmetric convolutional filters in inception modules. For example, using n x1 filters followed by 1×n filters are 33% computationally cheaper than n×n convolution filters (Szegedy et al., 2015). Pre-trained InceptionV3 can be directly applied to natural image classification tasks and used for TL purposes in other domains. The main reason for using this model for bone lesion classification is that InceptionV3 is robust against overfitting with limited data.

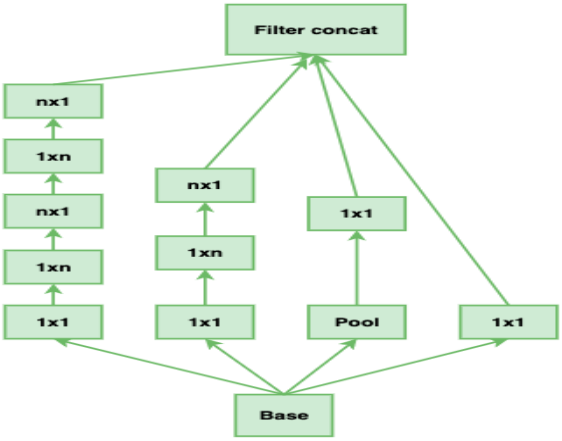


Figure 3: Inception module with n × n convolution. Adapted from: (Szegedy et al., 2015)

4.3.3 *ResNet50*

The Resnet50 model was developed in 2015 and consisted of 50 layers (K. He, Zhang, Ren, & Sun, 2015). One of the unique features of this model is the residual network architectures (Figure 4). Very Deep learning structures suffer from gradient vanishing or exploding problems, and the residual network design helps the model overcome this. In the residual network, while adding the activation to the next layer, the same activation will skip some layers and be mapped to two or three layers ahead. In this way, the extracted features will not vanish. Training time and cost are essential factors for deep learning models. In the Resnet50, the residual block is designed in the bottleneck approach, which allows the model to train faster. The bottleneck building block contains three convolutional layers with 1×1, 3×3, and 1×1 filters, and the 1×1 filter reduces the trainable parameters (K. He et al., 2015). Despite the deeper structure of Resnet50, it has fewer floating point operations (FLOPs) than shallow models like VGG19. Resnet50 is widely used in medical image classification. At the same time, the model trains fast while preventing the gradient from vanishing.

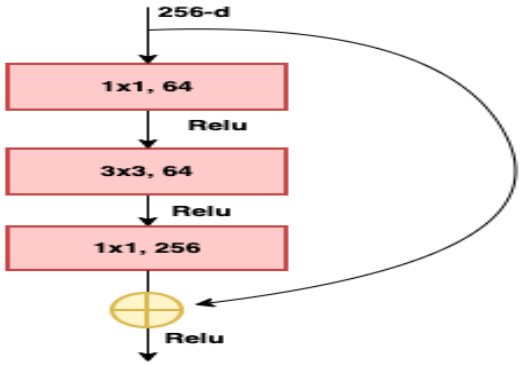


Figure 4: Building block of Resnet50. Adapted from (K. He et al., 2015)

4.3.4 *EfficientNetB7*

EfficienNetB7 is one of the new versions of the EfficientNet model family. This model is designed based on a new way to scale model dimensions like depth, width, and resolution (Tan & Le, 2019). First, the relationship of different scaling dimensions is searched based on the grid search to find an optimal value for the compound scaling method. Then, the model is scaled up according to the compound coefficient, balancing the network dimensions optimally (Figure 5). This model is relatively new, and there is limited study on EfficientNetB7 in medical image classification. However, it is a relatively large model with more than 66 million parameters and can achieve very high accuracy on image classification (Y. He et al., 2020). This factor makes it interesting to test in this study.

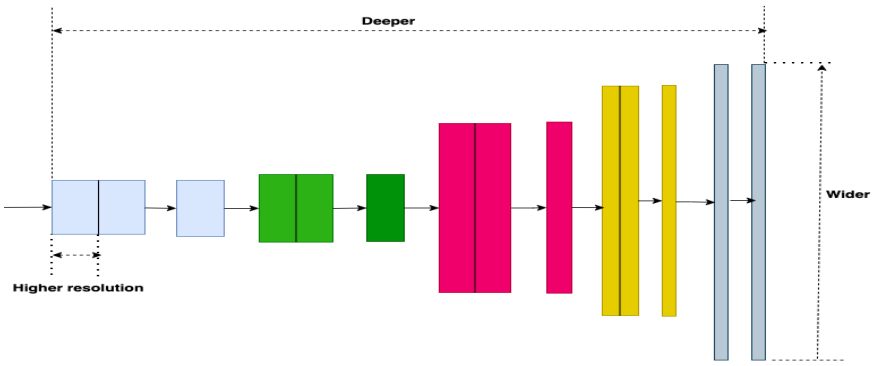


Figure 5: Model scaling methods in EfficientNetB7. Adapted from: (Tan & Le, 2019)

4.3.5 *Requirements for preprocessing images in pre-trained models*

The pre-trained models employed in this study were trained on different image input shapes. Omitting the fully-connected layers at the top of the model enables the use of different input shapes for the bone lesion classification. Furthermore, each pre-trained model necessitates a specific method for processing the input images 2. These preprocessing methods allow the pre-trained models to achieve the best performance. The TensorFlow framework offers a built-in preprocessing approach for each pre-trained model. These models can be trained to utilize ImageNet weights and subsequently fine-tuned to enhance their performances.

Model name	Pre-trained image shape	Preprocessing method
VGG16	224x224	Converted to BGR and zero-centered.
InceptionV3	299x299	[-1,1]
Resnet50	224x224	Converted to BGR and zero-centered.
EfficienNetB7	224x224	[0,255]

Table 2: Preprocessing requirements of pre-trained models

4.4 *Data augmentation*

Deep learning models heavily rely on large data to prevent overfitting. Data augmentation techniques will enhance the size and quality of the dataset such that deep learning models avoid overfitting. Furthermore, it is possible to apply data augmentation while the patches are being created. On the other hand, the TensorFlow framework provides excellent data augmentation tools that allow users to augment the data directly from the image directory while the model is training. This method makes it possible to find the best data augmentation techniques for the models. Table 3 shows the data augmentation techniques which are applied in this thesis.

Data augmentation type	Range
Rotation	[0,180]
Width shift	[0,1]
Height shift	[0,1]
Horizontal flip	True, False
Shear	[0,1]
Zoom	[0,1]
Fill mode	Nearest, constant, reflect, wrap

Table 3: Basic data augmentation techniques

4.5 *Performance metrics*

The performance metric depends on the type of machine learning problems. For example, accuracy, Precision, F1-score, recall, and specificity are mainly used for classification tasks. All these metrics can be calculated from the number of TPs, TNs, FPs, and False Negatives(FNs) (Appendix A, page 36). These four values can be calculated based on the built-in confusion matrix function of the scikit-learn library. Accuracy is mainly applied during the model training phase and is the ratio between all correctly classified values and the total number of samples from a dataset. It is the best metric to examine the model behavior during the training phase, revealing whether a model is overfitting or underfitting. When a dataset is not balanced, accuracy is not the right metric to evaluate the model. F1-score is the harmonic mean of precision and recall (Lipton, Elkan, & Narayanaswamy, 2014). For imbalanced data, it gives a more accurate measure than accuracy. Precision measures TPs against all positively predicted classes, including TPs and FPs. Recall computes the predicted TPs against actual TPs values from the dataset, including TPs and FNs. Since this thesis focuses on classifying the FPs, it is better to use the FPR and False Negative Rate(FNR) for model evaluation. These metrics can be easily recalculated from sensitivity/recall and specificity. FPR equals $FPR = 1 - specificity$, and FN rate is $FNR = 1 - Recall$. It is better to visualize the trade-off between TPs and FPs. The ROC curve is the best way to illustrate this across the threshold line. After each model training, the performance metrics are calculated on validation, test dataset, and final test set. In this way, it is possible to examine the generalizability of the models.

4.6 *Software and libraries*

The models will be developed and trained on the Linux server of the ETZ hospital. Python will serve as the primary programming language for this thesis, and the models will be constructed based on the TensorFlow framework. In addition, various python libraries such as NumPy, Scikit-image, and Scikit-learn will be applied for the data preprocessing.

4.7 *Implementation details*

4.7.1 *Model design*

The standard procedure for implementing TL involves removing the final classification layers of a pre-trained model and replacing them with a

self-designed classifier. This pre-trained model, called the base model, is utilized as a feature extractor. Commonly, the base model is followed by a Global Average Pooling (GAP) layer or a flatten layer. The flatten layer serves to transform any tensor shape into a one-dimensional tensor. However, it has been noted that the use of a flatten layer may increase the susceptibility to overfitting, particularly when the amount of training data is limited (Lin, Chen, & Yan, 2013). On the other hand, the GAP layer resists the overfitting problem by minimizing the parameters. This GAP layer takes a tensor with $h \times w \times d$ dimensions and averages each feature map, resulting in a factor of $1 \times 1 \times d$ (Figure 6).

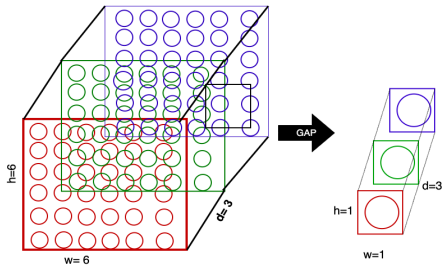


Figure 6: Visualization of Global average pooling

In this thesis, four pre-trained models will have the same final classifier, as shown in Figure 7. First, the final classification layers of pre-trained models are detached. Then, a GAP layer is attached to the pre-trained model; it is also possible to activate the GAP layer from a built-in parameter from the TensorFlow framework. After that, a dense layer is attached to the GAP layer, and a final output layer has two units with the softmax activation function. The number of units for the dense layers will be determined during the hyperparameters tuning. The softmax will return a vector with two numbers representing each class's prediction probability.

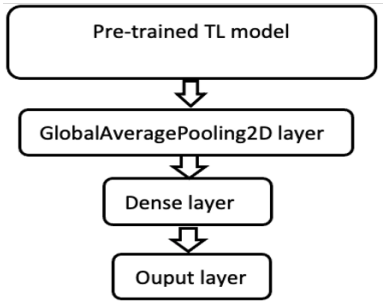


Figure 7: model design with base models

4.7.2 *Hyperparameters tuning*

Hyperparameter tuning is an essential aspect of training and validating a model. However, identifying the optimal values for these hyperparameters poses a significant challenge. Utilizing a Grid Search approach for DCNNs requires computational resources and time. Therefore, for this thesis, the selection of hyperparameters was based on manual testing and the utilization of built-in functions. A total of eight different hyperparameters with varying values were considered for each pre-trained model (Table 4). Due to the complexity of simultaneously searching for more than two hyperparameters, only one or two hyperparameters were adjusted and evaluated during each training session until the improved performance was achieved. Additionally, the TensorFlow framework's ReduceLROnPlateau function allows for the utilization of different learning rates within a specified range. The ReduceLROnPlateau function enables the model to automatically reduce the learning rate during training and determine the most appropriate learning rate. As previously discussed in the model design section, implementing GAP is particularly beneficial for models with limited datasets, as it reduces the number of trainable parameters and acts as a regularizer to prevent overfitting.

A pre-trained model with newly added dense layers has only a few learnable weights, mainly from the dense layer. Increasing the number of units in a dense layer causes the model to overfit easily on limited data (Garbin, Zhu, & Marques, 2020). After trying different unit sizes, the unit size of 64 resulted in higher performance for all models. The batch size determines the number of examples processed by the model before updating the trainable parameters. Finding the best tradeoff between batch size, learning rate, and optimizers takes a lot of work. In the end, these factors determine the rate and size of the gradient updates (Wilson & Martinez, 2003). In order to train the model without overfitting, a range of batch sizes are tested, and a batch size of 8 with a minimal learning rate between 0.00001 and 0.000001 resulted in the best performance.

Adam optimizer is a stochastic gradient-based optimizer based on adaptive estimation of first-order and second-order moments (Kingma & Ba, 2014). Adam needs less memory and is computationally efficient. On the other hand, the Nadam is the Adam optimizer with Nesterov momentum that reduces oscillations of noisy gradients. After testing both optimizers, the Adam optimizer performed better than the Nadam optimizer. For binary classification, it is possible to use binary cross-entropy or categorical cross-entropy. However, it differs from the final output player of a model, sigmoid activation function for binary cross-entropy, and softmax activation for categorical cross-entropy. Besides that, there is no performance difference between these loss functions. Furthermore,

the epochs of the model are set to 40 with an early stopping function, and all models converged between 20 and 30 epochs. Finally, the models are trained with pre-trained weights and fine-tuned by unfreezing the trainable layers.

Hyper-params	**Values range**	**Best option**	**Help function**
Learning rate	[0.01, 0.0000001]	Determined by ReduceLROnPlat-eau	ReduceLROnPlateau
Dense layer	Flatten () GlobalAveragePooling2D()	GlobalAveragePooling2D()	None
Units in dense layers	[32,64,128,512]	64	None
Batch size	[4,8,16,32,64]	8	None
Optimizers	Adam, Nadam	Adam	Keras. Optimizers.Adam
Loss functions	Binary cross entropy Categorical crossentropy	CategoricalCrossentropy	None
Epochs	[10,20,40]	40	EarlyStopping
Freeze and unfreeze layers	True, False	True	model.trainable

Table 4: Hyperparameters of models with selected values

4.7.3 *Model training process*

It is essential to clarify the process of model training. As described in the previous section, pre-trained models are attached with an average pooling layer with a dense layer. Since each pre-trained model was trained with a specific image resolution, it is important to investigate the effect of different image resolutions; therefore, the training process began with training the baseline model, VGG16, with different image resolutions (Figure 8). After that, the best image resolution is selected based on the performance of VGG16. The pre-trained models are first trained with frozen layers. The pre-trained models are fine-tuned in the following training sessions by unfreezing all the hidden layers. After that, all models are tested on the test set and hold-out test set, which consists of FPs samples from the segmented bone lesions. Finally, an ensemble learning method combining the predictions applied for all models and tested on a hold-out test set.

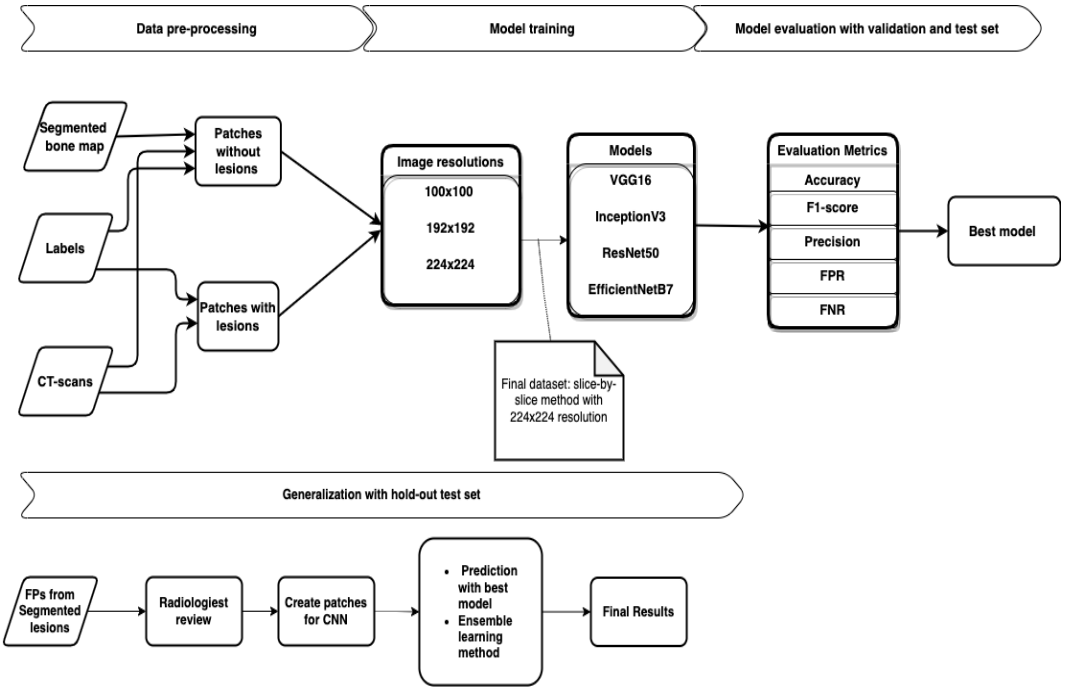


Figure 8: The flow chart of the thesis

5 RESULT

5.1 *Selection of image Resolution for models*

It is challenging to test the performance of the pre-trained model on different image resolutions and data augmentation techniques at a time. Therefore, the baseline model is gradually tested on different image resolutions and data augmentation techniques (Figure 8).

The pre-trained VGG16 model was trained on the three different image resolutions without data augmentation. The final result indicates that VGG16 on the image resolution of 224×224 outperformed the datasets with other image resolutions. Therefore, the next step of the training phase was continued with a dataset with 224×224 image resolution.

In the next step, the VGG16 is trained with different combinations of data augmentation methods. The model with more data augmentation methods overfitted even more than without data augmentation methods. After running the model a couple of times with different combinations, the final data augmentation combination consists of rotation and horizontal flips, shifting with the reflected fill method. One of the interesting areas in data augmentation is the filling mode for augmented images. When an image is augmented with rotation or shifting, the image will get an

empty blank area. The filling method will fill this spot with different techniques like contracting and reflecting(Figure 9). After testing the VGG16 model with different filling methods, the 'reflect' filling resulted in higher validation accuracy.

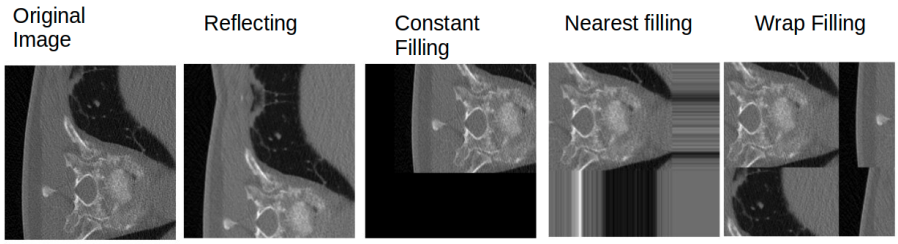


Figure 9: Different types of filling methods in TensorFlow

Table 5 shows the result of VGG16 on three different image resolutions, with and without pre-trained weights. It is important to report that the values in Table 5 are given in weighted average. The image resolution of 224x 224 resulted in a validation accuracy of 0.86 and outperformed the dataset with other image resolutions. After that, the dataset with 224×224 image resolution is trained with data augmentation with a rotation range of 90, horizontal flipping, and a shifting range of 0.2. The output showed that the data augmentation reduces the overall performances slightly; however, the model is less overfitted than the model without data augmentation (Figure 10). The pre-trained InceptionV3 model was also trained with and without data augmentation. This ensures the effectiveness of the data augmentation techniques for DCNN models; the final result indicated that data augmentation prevents the model from overfitting (Figure 10). The pre-trained DCNN models have been observed to be sensitive to overfitting without data augmentation. As a result, data augmentation is essential for these models. However, due to time and computational resource limitations, it was not feasible to test the other two models in this experimental session.

pre-trained model	Data augmentation	Image resolutions	Val- acc	Precision	F1-score	FPR	FNR
VGG16	No	100X100	0.77	0.77	0.77	0.25	0.21
	No	192x192	0.81	0.82	0.81	0.26	0.11
	No	224x224	0.86	0.86	0.86	0.17	0.11
	Basic data augmentation	224x224	0.79	0.79	0.79	0.18	0.25

Table 5: The result of baseline model(VGG16)

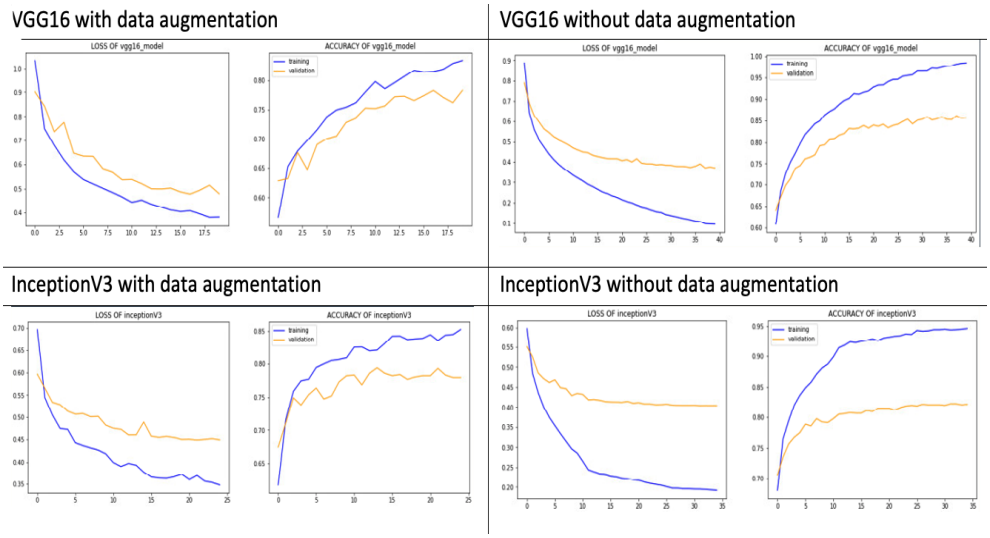


Figure 10: VGG16 and InceptionV3 with and without data augmentation

5.2 *Comparison of VGG16 with and without pre-trained weights*

In order to examine the effect of pre-trained weights, the VGG16 was also trained with randomly initialized weights. The result indicated that pre-trained VGG16 performed better than VGG16 with randomly initialized weights. Overall, the F-1 score of pre-trained VGG16 is 15% higher than the VGG16 with randomly initialized weights. At the same time, the FPR and FNR of the pre-trained model are relatively lower than the other model, so the ImageNet weights can extract image features better than the randomly initialized weights.

Model	**Val- acc**	**Precision**	**F1-score**	**FPR**	**FNR**
pre-trained VGG16	0.79	0.79	0.79	0.18	0.25
VGG16 with randomly initialized weights	0.64	0.64	0.64	0.34	0.37

Table 6: Effect of pre-trained and randomly initialized weights on VGG16

5.3 *Comparison of pre-trained models*

The pre-trained models are trained on a dataset with 224×224 image resolution with basic data augmentation. Table 7 shows the result of pre-trained models. Figure 11 showed that the ResNet50 and Effi-cientNetB7

performed better than the InceptionV3 and VGG16. Both Resnet50 and EfficientNetB7 achieved an accuracy of 0.82 on the validation dataset. The FPR of these two models is 0.16, while InceptionV3 and VGG16 have an FPR of 0.21. FNR of ResNet50 and EfficientNetB7 are 0.19 and 0.21, respectively.

Pre-trained models on the validation dataset					
Models	Val-acc	precision	F1-score	FPR	FNR
VGG16	0.77	0.77	0.77	0.21	0.26
InceptionV3	0.78	0.78	0.78	0.21	0.23
ResNet50	0.82	0.83	0.83	0.16	0.19
EfficientNetB7	0.82	0.82	0.82	0.16	0.21

Table 7: The results of pre-trained models on validation set

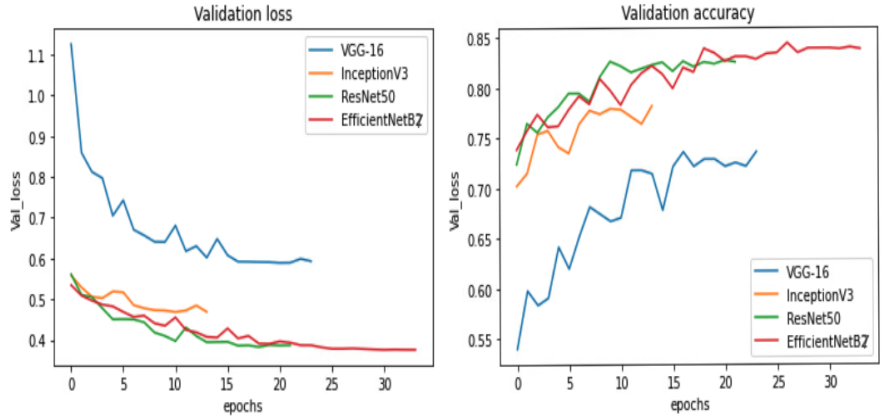


Figure 11: Training accuracy and loss of pre-trained models

The models achieved a higher performance when their hidden layers were unfrozen (fine-tuned). After the fine-tuning, ResNet50 and EfficientNetB7 achieved a validation accuracy of 0.93 and 0.96, respectively (Table 8. In addition, these two models resulted in very low FPR and FNR.

Fine-tuned Models on the validation dataset					
Models	Val-acc	precision	F1-score	FPR	FNR
VGG16	0.91	0.91	0.91	0.08	0.05
InceptionV3	0.92	0.92	0.92	0.08	0.08
ResNet50	0.93	0.93	0.93	0.07	0.06
EfficientNetB7	0.96	0.96	0.96	0.05	0.03

Table 8: The results of fine-tuned models on the validation set

The test dataset results indicated that fine-tuned EfficientNetB7 outperformed the all-other models, reaching a test accuracy of 0.93, a precision score of 0.94, and an F1-score of 0.93 (Table 9). The EfficientNetB7 also had an FPR of 0.12 and an FNR of 0.02. However, the results showed that all models have difficulty classifying the non-lesions, while these models can achieve higher recall for lesion class (Appendix B, page 37). Figure 12 illustrates the confusion matrix and ROC curve of the EfficientNetB7. The number of FPs and FNs is very low, and the ROC curve reveals that both classes have high TPR.

Fine-tuned models on the test dataset					
Models	Val-acc	precision	F1-score	FPR	FNR
VGG16	0.91	0.92	0.91	0.14	0.03
InceptionV3	0.91	0.92	0.91	0.14	0.04
ResNet50	0.91	0.91	0.91	0.12	0.05
EfficientNetB7	0.93	0.94	0.93	0.12	0.02

Table 9: The results of fine-tuned models on the test set

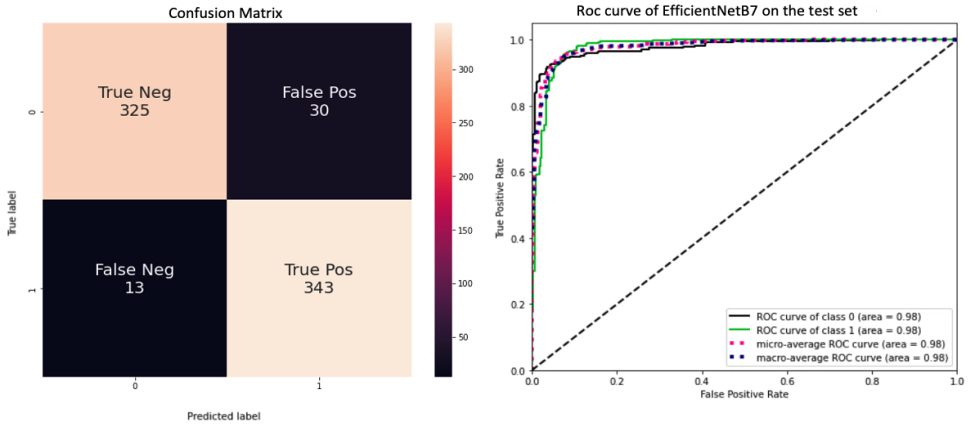


Figure 12: Confusion Matrix and ROC curve of EfficientNetB7 on the test dataset

5.4 *Generalization with hold-out test set*

As the research question indicated, the final goal of this thesis was to develop a DCNN model to classify the FPs from segmented bone lesions. Therefore, the radiologist reviewed a few patients' segmented bone lesions, resulting in the hold-out test set with 245 samples. These samples were initially predicted as FPs by the 2D U-net(segmentation algorithm). Eighteen FPs of this dataset are labeled as TPs, and the other 277 images are labeled with non-lesion (FPs) related labels like nothing, fat, intervertebral disc, etc.

Table 10 gives the final result of four models on the hold-out test set. Because this data set is imbalanced, all values are given in weighted averages. Overall, the InceptionV3 and EfficientNetB7 performed better than VGG16 and ResNet50. However, the ResNet50 was good at classifying the TPs, and it classified all TPs samples correctly. Finally, the EfficientNetB7 outperformed all other models and has a weighted F1-score of 0.69 with an FPR of 0.42.

The fine-tuned models on hold-out test set					
Models	Val-acc	precision	F1-score	FPR	FNR
VGG16	0.34	0.92	0.43	0.7	0.06
InceptionV3	0.54	0.88	0.65	0.46	0.39
ResNet50	0.32	0.93	0.4	0.74	0.0
EfficientNetB7	0.6	0.92	0.69	0.42	0.17

Table 10: The results of fine-tuned models on the hold-out test set

Since EffientNetB7 has the highest overall score, its output is analyzed more deeply. From Figure 13 can be seen that 95 of 227 non-lesions (TNs) are still classified as lesions(FPs). 3 out 18 lesions(TPs) are classified as non-lesions(FNs). ROC curves contain more zigzag patterns, which is caused by the limited samples. Nevertheless, the model on the hold-out test set did not perform as good as on the test and validation dataset. This may be attributed to the method employed for extracting the patches. The location of bones on non-lesion patches in the training dataset is randomly extracted; however, actual FPs predictions in the hold-out test set are cropped around FPs(centered). The EfficientNetB7 is retested on a hold-out test set with data augmentation methods like width and height shifting range of 0.2. Interestingly, the number of FPs from prediction dropped to 49. So, randomizing the locations of the bone on the patches will increase the model performance; however, this caused an increase in the FNs.

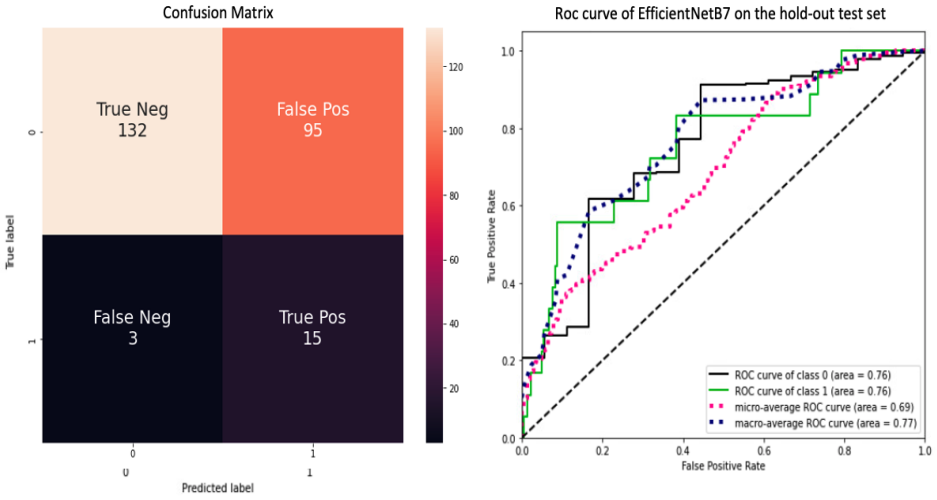


Figure 13: confusion matrix and ROC curve of EfficientNetB7 on the hold-out test set

Figure 14 illustrates the misclassified images from the hold-out test set. Patch numbers from 1 to 4 are the non-lesions that are still misclassified as lesions by EfficietNetB7. The patches, 5,6,7 are actual lesions(TPs) that are misclassified as non-lesions(FNs) by the model. Based on these images, it could be concluded that complex bone patterns and metal artifacts (patch-7) could also be reasons for misclassification.

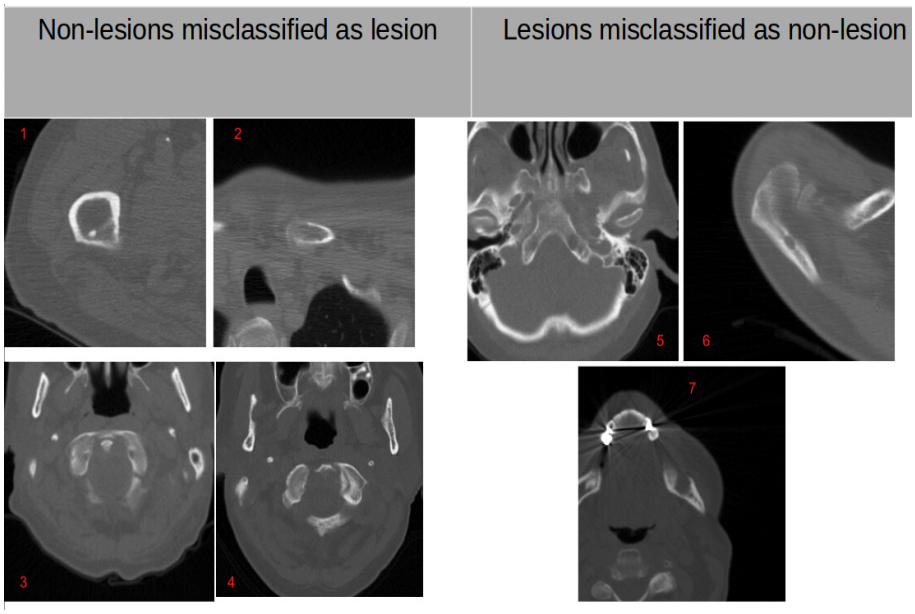


Figure 14: The sample images which are misclassified from EfficientNetB7

5.5 *Comparison with ensemble learning on hold-out test set*

As Table 10 described, the ResNet50 is good at classifying TPs, so in the Next test, the prediction probabilities of ResNet50 and the EfficientNetB7 are combined, and the average prediction values are retrieved. This method is also known as Stacking in ensemble learning ((Müller et al., 2022)). As the results indicated in Table 11, combining the prediction probabilities of the two models increases the overall performances. The ensemble model resulted in an F1-score of 0.83, an FPR of 0.17, and an FNR of 0.67. Finally, the same stacking method is applied for all four models, so the final predictions are the stacked and averaged of these four models. The result shows that combining all models is better for reducing the FPs and FNs. The FPR dropped from 0.17 to 0.15, and FNR decreased from 0.67 to 0.5.

The results of combined models on the final test set					
Combined models	Val-acc	precision	F1-score	FPR	FNR
ResNet50 + EfficientNetB7	0.79	0.88	0.83	0.17	0.67
VVG16+InceptionV3+ ResNet50 + EfficientNetB7	0.82	0.90	0.85	0.15	0.5

Table 11: The results of ensemble learning

6 DISCUSSION

The main goal of this thesis was to develop a DCCN model for classifying the FPs from segmented bone lesions caused by MM disease. The ETZ hospital provides the dataset for this thesis, and the dataset contains more than 97 CT scans from 79 patients. The main research question was, "To what extent can DCNN models classify potential FPs or missed lesion candidates in segmented osteolytic bone lesions?". After reviewing the previous studies, it was clear that the TL is most suitable for this task because pre-trained models are effective on limited data. The NIFTI file can not be directly used in DCNN models with 2DConv layers. In the first step of this thesis, three different image resolutions are compared to each other, and the outcome will be discussed in section 6.1. Furthermore, section 6.2 will review the result of the pre-trained and fine-tuned models. After that, model generalization on the hold-out test set the effect of ensemble learning techniques is discussed in sections 6.3 and 6.4. Finally, the limitations of this study are given in section 6.4.

6.1 *Image resolutions*

DCNN models with 2D convolutional layers require 2D images that are either single or multi-channel. Since the CT scans are in 3D NIFTI format, they cannot be directly fed into DCNN models with 2D Conv layers. Therefore, two different patch types are cropped and extracted from the CT scans. Furthermore, all patches are saved as three-channeled RGB images because three-channeled images result in better model performances than one-channeled images (Perkonigg et al., 2018).

The baseline model, VGG16, is trained on three different image resolutions in the first training phase. The result showed that the image resolution of 224×224 resulted in a better performance on the validation dataset. Previously thought that the model could learn more with a smaller resolution because of the larger proportion of the lesion on a patch, but a model with smaller patches is not good as a model with larger patches. The experiment demonstrated that higher resolution resulted in better performance; however, there is a maximum boundary that DCNN models cannot enhance their performance as image resolution increases. At the same time, DCNN models with larger images need more computational power, which is not cost-efficient. According to Sabottke and Spieler (2020), the limit of image resolution on model performance is 224×224. Interestingly, the VGG16, ResNet50, and EfficietNetB7 were initially trained on 224x224 (K. He et al., 2015; Simonyan & Zisserman, 2014; Tan & Le, 2019).

As a result, the three channeled patches with 224x224 resolutions are the best for this bone lesion classification task.

6.2 *Best pre-trained model*

The first subquestion was, "To what extent can TL increase classification model performance?". In order to answer this question, four different pre-trained models are trained and fine-tuned. The final results showed that fine-tuned models resulted in better model performance. The EfficientNetB7 outperformed the other three models and reached an F1-score of 0.69 on the hold-out test dataset.

The VGG16 model has shallower architectures than the other three models, but it achieved a test accuracy score of 0.91 on the test dataset with PFRs of 0.14 and FNR of 0.03. As Yan et al. (2018) indicated, VGG16 is one of the best models to use as a baseline for medical images. Despite its ability to resist overfitting (Szegedy et al., 2015), InceptionV3 did not perform as expected. During the training, this model stopped earlier than the other three (Figure 11). The reason could be the hyperparameters or the image resolution since InceptionV3 was pre-trained on 299×299 input shapes.

Raghu et al. (2019) indicated that ResNet50 achieved very high accuracy on eye-related medical images; however, ResNet50 did not perform as expected on this bone lesion classification. One of the reasons could be the available dataset size since (Raghu et al., 2019) trained the ResNet50 on a very large dataset with 100,000 images.

EfficientNetB7, as outlined in the recent study by Tan and Le (2019), is a state-of-the-art pre-trained model. The results of this evaluation indicate that it outperforms other models in classifying FPs. This finding is supported by previous research on bone cancer detection, which achieved high performance using an older model version, EfficientNetB0. These results demonstrate the effectiveness of EfficientNetB7 for bone lesion classification.

The final results indicate that fine-tuning models yield higher performance than pre-trained models with ImageNet weights. This finding is consistent with the proposal of Raghu et al. (2019) that a deep pre-trained model may not perform any better than shallow models, such as VGG16, in bone lesion classification. However, our results demonstrate that fine-tuning the model by unfreezing the layers yields the best results.

6.3 *Generalization with hold-out test set*

The second subquestion addressed in this thesis was, "To what extent can DCNN models generalize to the dataset created from the feedback of radiologists?". The results of the fine-tuned EfficientNetB7 model on the hold-out test set were inferior compared to its performance on the test set. Specifically, the F1 score of the fine-tuned EfficientNetB7 model on the test set was 0.93; however, it dropped to 0.69 on the hold-out test set. Analysis of the hold-out test set revealed that 95 out of 227 FPs were classified as FPs. This higher number of FPs could be attributed to the location of the bone in the patches, as the non-lesion patches in the training dataset were randomly generated. However, the FP samples from the final test set were cropped around the predictions of the U-net model. In order to address this issue, the final model was retested using data augmentation techniques such as width and height shifting. The results of this retesting showed that the number of FPs dropped to 46; however, this resulted in an increase in FNR. Further research is required to investigate methods of balancing the trade-off between FPR and FNR.

6.4 *Effect of ensemble learning*

The third subquestion was: "To what extent can ensemble learning improve the classification result?" The utilization of ensemble learning with both ResNet50 and EfficientNetB7 models revealed an increase in the F1 score from 0.69 to 0.83. A decrease in FPR was also observed; however, this resulted in an increase in FNR. This finding is consistent with the research conducted by Müller et al. (2022), who demonstrated that ensemble learning, which involves stacking the prediction probabilities of multiple models, can improve overall performance in classification tasks.

The results of ensemble learning with four models indicated that it could effectively improve the classification of FPs. Furthermore, combining the prediction probabilities of four fine-tuned models resulted in slightly better performance than combining the output of only two models (Table 11). Interestingly, the FPR and FNR in this ensemble model are further decreased than in the ensemble model with ResNet50 and EfficientNetB7 alone.

6.5 *Limitations*

As discussed throughout this thesis, a significant limitation of this study is the availability of limited annotated data. The identified lesions from CT scans are scarce, and the use of open-source data for the classification

of bone lesions is challenging due to the lack of radiologist expertise to assess the dataset's quality. Furthermore, the lesions are distributed throughout the entire body, presenting a high degree of variance in shape and appearance.

Additionally, determining optimal values for hyperparameters is complex, particularly when limited training sessions are conducted. The results of this study indicate that the selection of appropriate hyperparameters directly impacts model performance. Although each model in this thesis was trained with the same hyperparameters, different settings and values may be more appropriate given the distinct designs of each model.

Another limitation is the effective implementation of data augmentation techniques. Due to the limited time and computational cost, only basic data augmentation techniques were tested in this thesis. Furthermore, there is a paucity of research on the effects of data augmentation on the classification of bone lesions.

7 RECOMMENDATION

The results of this thesis revealed that the pre-trained models did not perform as expected on the bone lesion dataset. However, fine-tuning these models led to better performance. This is likely due to the limited data, which caused the DCNN models in this thesis to be unable to extract sufficient features. To address this limitation, one possible solution is to train a model on publicly available medical data and then apply this pre-trained model as TL for bone lesion classification. This approach would allow the medical image weights of the model to be effectively utilized for bone lesion classification.

The grid search algorithm represents a simple approach to hyperparameter tuning. However, its utilization in DCNN requires significant computational resources and time. Therefore, an alternative method, such as a random search utilizing the KerasTuner framework, maybe a more efficient solution. KerasTuner, a scalable hyperparameter optimization framework developed for the Keras library, offers ease of implementation (O'Malley et al., 2019).

In addition, the results of the final test set indicate that the method of patch extraction significantly impacts model performance. To mitigate this effect, patches should be extracted from random locations, as this will ensure that lesions or non-lesions are present in random locations within the patches, thereby improving the robustness of the model. Another consideration is the utilization of ensemble deep learning methods. The combined result of ResNet50 and EfficientNetB7 has demonstrated the effectiveness of ensemble methods in improving performance. Furthermore,

ensemble learning with multiple models reduced both the FPR and FNR. Therefore, it is likely that other types of ensemble learning methods will further increase the classification performance.

8 CONCLUSION

The primary objective of this study was to develop a DCNN model for the classification of FPs from segmented bone lesions. Accordingly, the research question addressed was, "To what extent can DCNN models classify potential PFs or missed lesion candidates in segmented osteolytic bone lesions?". The final results of a series of experiments indicated that TL is a suitable approach for bone lesion classification. Specifically, when the models were fine-tuned and retrained with basic data augmentation techniques, the EfficientNetB7 model achieved an F1-score of 0.93 on the test set and 0.69 on the heavily imbalanced hold-out test set. Additionally, the results of the EfficientNetB7 on the hold-out test set revealed that the fine-tuned EfficientNetB7 could reduce the number of FPs from the segmented bone lesions. The ensemble learning method was also proven effective in reducing FPs and FNs.

These outcomes suggest that it is feasible to train bone lesion classifiers using pre-trained DCNN models on limited datasets. However, the final results indicate that the model is not yet robust enough for practical settings. Pre-trained DCNN models heavily rely on large amounts of training data, patch types, and data augmentation methods. To enhance model performance, further research is needed to investigate the recommendations of this thesis.

REFERENCES

Deepak, S., & Ameer, P. M. (2019, 8). Brain tumor classification using deep cnn features via transfer learning. *Computers in Biology and Medicine*, *111*. doi: 10.1016/j.compbiomed.2019.103345

Eweje, F. R., Bao, B., Wu, J., Dalal, D., hua Liao, W., He, Y., ... States, L. (2021, 6). Deep learning for classification of bone lesions on routine mri. *EBioMedicine*, *68*. doi: 10.1016/j.ebiom.2021.103402

Filho, A. G., Carneiro, B. C., Pastore, D., Silva, I. P., Yamashita, S. R., Consolo, F. D., ... Nico, M. A. (2019, 7). Whole-body imaging of multiple myeloma: Diagnostic criteria. *Radiographics*, *39*, 1077-1097. doi: 10.1148/rg.2019180096

Garbin, C., Zhu, X., & Marques, O. (2020, 5). Dropout vs. batch normalization: an empirical study of their impact to deep learning. *Multimedia Tools and Applications*, *79*, 12777-12815. doi: 10.1007/s11042-019-08453-9

Gouillart, E., Nunez-Iglesias, J., & Walt, S. V. D. (2017). Analyzing microtomography data with python and the scikit-image library. *Advanced Structural and Chemical Imaging*, 18. Retrieved from https://github.com/jni/python-redshirt (Image prop and connected components) doi: 10.1186/s40679-016-0031-0Ãŕ

He, K., Zhang, X., Ren, S., & Sun, J. (2015, 12). Deep residual learning for image recognition. Retrieved from http://arxiv.org/abs/1512.03385

He, Y., Pan, I., Bao, B., Halsey, K., Chang, M., Liu, H., ... Bai, H. X. (2020, 12). Deep learning-based classification of primary bone tumors on radiographs: A preliminary study. *EBioMedicine*, *62*. doi: 10.1016/j.ebiom.2020.103121

Hossain, M. B., Iqbal, S. M. S., Islam, M. M., Akhtar, M. N., & Sarker, I. H. (2022, 1). Transfer learning with fine-tuned deep cnn resnet50 model for classifying covid-19 from chest x-ray images. *Informatics in Medicine Unlocked*, *30*. doi: 10.1016/j.imu.2022.100916

Kingma, D. P., & Ba, J. (2014, 12). Adam: A method for stochastic optimization. Retrieved from http://arxiv.org/abs/1412.6980

Koshiaris, C. (2019, 2). Methods for reducing delays in the diagnosis of multiple myeloma. *International Journal of Hematologic Oncology*, *8*, IJH13. doi: 10.2217/ijh-2018-0014

Lin, M., Chen, Q., & Yan, S. (2013, 12). Network in network. Retrieved from http://arxiv.org/abs/1312.4400

Lipton, Z. C., Elkan, C., & Narayanaswamy, B. (2014, 2). Thresholding classifiers to maximize f1 score. Retrieved from http://arxiv.org/abs/1402.1892

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., ... Sánchez, C. I. (2017, 12). A survey on deep learning in medical image analysis. *Medical Image Analysis*, *42*, 60-88. doi: 10.1016/j.media.2017.07.005

Maskell, G., & Frcp, F. (2019). *Commentary error in radiology-where are we now?*

Mohammed, M., Mwambi, H., Mboya, I. B., Elbashir, M. K., & Omolo, B. (2021). A stacking ensemble deep learning approach to cancer type classification based on tcga data. *Scientific reports*, *11*(1), 1–22.

Müller, D., Soto-Rey, I., & Kramer, F. (2022). An analysis on ensemble learning optimized medical image classification with deep convolutional neural networks. *arXiv preprint arXiv:2201.11440*.

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., et al. (2019). *Kerastuner.* https://github.com/keras-team/keras-tuner.

Perkonigg, M., Hofmanninger, J., Menze, B., Weber, M. A., & Langs, G. (2018). Detecting bone lesions in multiple myeloma patients using transfer learning. In (Vol. 11076 LNCS, p. 22-30). Springer Verlag. doi: 10.1007/978-3-030-00807-9_3

Raghu, M., Zhang, C., Kleinberg, J., & Bengio, S. (2019, 2). Transfusion: Understanding transfer learning for medical imaging. Retrieved from http://arxiv.org/abs/1902.07208

Rajkumar, S. V., & Kumar, S. (2016, 1). Multiple myeloma: Diagnosis and treatment. *Mayo Clinic Proceedings*, *91*, 101-119. doi: 10.1016/j.mayocp.2015.11.007

Reagan, M. R., Liaw, L., Rosen, C. J., & Ghobrial, I. M. (2015, 6). Dynamic interplay between bone and multiple myeloma: Emerging roles of the osteoblast. *Bone*, *75*, 161-169. doi: 10.1016/j.bone.2015.02.021

Rister, B., Yi, D., Shivakumar, K., Nobashi, T., & Rubin, D. L. (2020, 12). Ct-org, a new dataset for multiple organ segmentation in computed tomography. *Scientific Data*, *7*. doi: 10.1038/s41597-020-00715-8

Ronneberger, O., Fischer, P., & Brox, T. (2015, 5). U-net: Convolutional networks for biomedical image segmentation. Retrieved from http://arxiv.org/abs/1505.04597

Sabottke, C. F., & Spieler, B. M. (2020, 1). The effect of image resolution on deep learning in radiography. *Radiology: Artificial Intelligence*, *2*. doi: 10.1148/ryai.2019190015

Schoninger, S., Homsi, Y., Kreps, A., & Milojkvovic, N. (2018, 10). A case of multiple myeloma misdiagnosed as seronegative rheumatoid arthritis and review of relevant literature. *Case Reports in Rheumatology*, *2018*, 1-5. doi: 10.1155/2018/9746241

Setio, A. A. A., Ciompi, F., Litjens, G., Gerke, P., Jacobs, C., Van Riel, S. J., ... Van Ginneken, B. (2016). Pulmonary nodule detection in ct images:

false positive reduction using multi-view convolutional networks. *IEEE transactions on medical imaging*, *35*(5), 1160–1169.

Simonyan, K., & Zisserman, A. (2014, 9). Very deep convolutional networks for large-scale image recognition. Retrieved from http://arxiv.org/abs/1409.1556

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015, 12). Rethinking the inception architecture for computer vision. Retrieved from http://arxiv.org/abs/1512.00567

Tan, M., & Le, Q. V. (2019, 5). Efficientnet: Rethinking model scaling for convolutional neural networks. Retrieved from http://arxiv.org/abs/1905.11946

Tong, S. (2021). *Automatix segmentation of the vertebrae in 3d ct scans using deep learning models.*

Wilson, D. R., & Martinez, T. R. (2003). The general inefficiency of batch training for gradient descent learning. *Neural Networks*, *16*, 1429-1451. doi: 10.1016/S0893-6080(03)00138-2

Xu, L., Tetteh, G., Lipkova, J., Zhao, Y., Li, H., Christ, P., ... Menze, B. H. (2018). Automated whole-body bone lesion detection for multiple myeloma on 68 ga-pentixafor pet/ct imaging using deep learning methods. *Contrast Media and Molecular Imaging*, *2018*. doi: 10.1155/2018/2391925

Yan, K., Wang, X., Lu, L., & Summers, R. M. (2018, 7). Deeplesion: automated mining of large-scale lesion annotations and universal lesion detection with deep learning. *Journal of Medical Imaging*, *5*, 1. doi: 10.1117/1.jmi.5.3.036501

Yang, Y., Hu, Y., Zhang, X., & Wang, S. (2022). Two-stage selective ensemble of cnn via deep tree training for medical image classification. *IEEE Transactions on Cybernetics*, *52*(9), 9194-9207. doi: 10.1109/TCYB.2021.3061147

Yang, Z., Chen, M., Kazemimoghadam, M., Ma, L., Stojadinovic, S., Timmerman, R., ... Gu, X. (2022). Deep-learning and radiomics ensemble classifier for false positive reduction in brain metastases segmentation. *Physics in Medicine & Biology*, *67*(2), 025004.

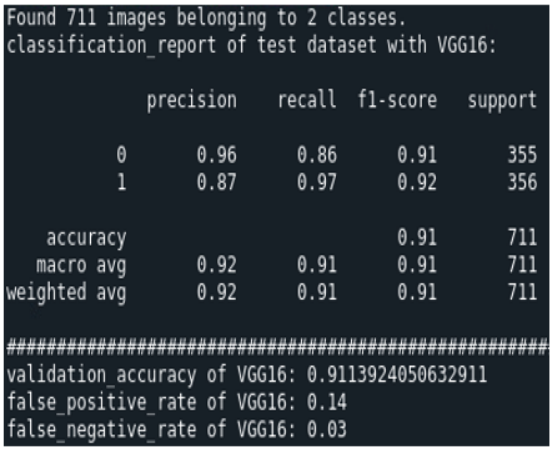
Zhao, D., Liu, Y., Yin, H., & Wang, Z. (2022). A novel multi-scale cnns for false positive reduction in pulmonary nodule detection. *Expert Systems with Applications*, 117652.

APPENDIX

APPENDIX A:PERFORMANCE METRICS

Performance metrics
Recall, Sensitivity, TPR $= \frac{TP}{TP+FN} = 1 - FNR$
$False\ Positive\ Rate = \frac{FP}{TN + FP}$
$Specificity, True\ Negative\ Rate\ (TNR) = \frac{TN}{TN+FP} = 1 - FPR$
$Precision = \frac{TP}{TP + FP}$
$False\ Negative\ Rate\ (FNR) = \frac{FN}{TP + FN}$
$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
$F1\ score = 2 * \frac{Precision * Recall}{Precision + Recall}$

APPENDIX B: CLASSIFICATION REPORTS OF MODELS



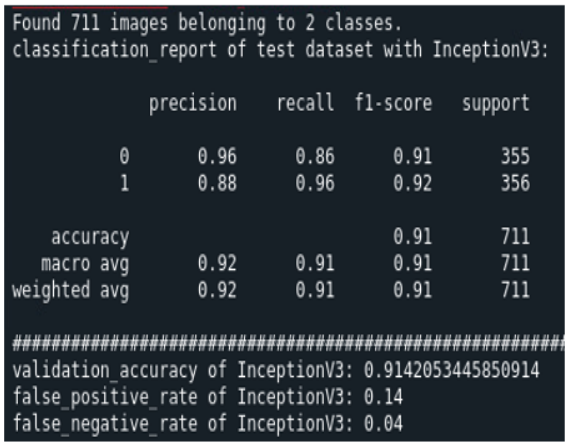
```
Found 711 images belonging to 2 classes.
classification_report of test dataset with VGG16:

              precision    recall  f1-score   support

           0       0.96      0.86      0.91       355
           1       0.87      0.97      0.92       356

    accuracy                           0.91       711
   macro avg       0.92      0.91      0.91       711
weighted avg       0.92      0.91      0.91       711

##################################################
validation_accuracy of VGG16: 0.9113924050632911
false_positive_rate of VGG16: 0.14
false_negative_rate of VGG16: 0.03
```



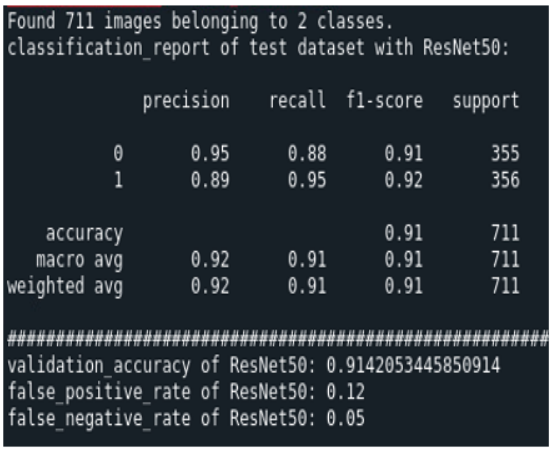
```
Found 711 images belonging to 2 classes.
classification_report of test dataset with InceptionV3:

              precision    recall  f1-score   support

           0       0.96      0.86      0.91       355
           1       0.88      0.96      0.92       356

    accuracy                           0.91       711
   macro avg       0.92      0.91      0.91       711
weighted avg       0.92      0.91      0.91       711

##################################################
validation_accuracy of InceptionV3: 0.9142053445850914
false_positive_rate of InceptionV3: 0.14
false_negative_rate of InceptionV3: 0.04
```



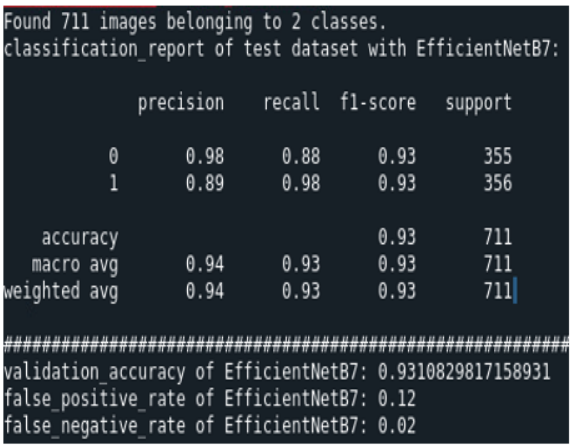
```
Found 711 images belonging to 2 classes.
classification_report of test dataset with ResNet50:

              precision    recall  f1-score   support

           0       0.95      0.88      0.91       355
           1       0.89      0.95      0.92       356

    accuracy                           0.91       711
   macro avg       0.92      0.91      0.91       711
weighted avg       0.92      0.91      0.91       711

##################################################
validation_accuracy of ResNet50: 0.9142053445850914
false_positive_rate of ResNet50: 0.12
false_negative_rate of ResNet50: 0.05
```



```
Found 711 images belonging to 2 classes.
classification_report of test dataset with EfficientNetB7:

              precision    recall  f1-score   support

           0       0.98      0.88      0.93       355
           1       0.89      0.98      0.93       356

    accuracy                           0.93       711
   macro avg       0.94      0.93      0.93       711
weighted avg       0.94      0.93      0.93       711

##################################################
validation_accuracy of EfficientNetB7: 0.9310829817158931
false_positive_rate of EfficientNetB7: 0.12
false_negative_rate of EfficientNetB7: 0.02
```