Mobile Computer-Aided Diagnosis (CAD) for Breast Cancer Detection

A Thesis

Presented in Partial Fulfillment of the Requirements for the

Degree of Master of Science

with a

Major in Computer Science

in the

College of Graduate Studies

University of Idaho

by

Jiaqiao Shi

Major Professors: Min Xian, Ph.D.

Committee Members: Aleksandar Vakanski, Ph.D.; Xiaogang Ma, Ph.D.

Department Administrator: Terry Soule, Ph.D.

May 2021

Authorization to Submit Thesis

This thesis of Jiaqiao Shi, submitted for the degree of Master of Science with a Major in Computer Science and titled "Mobile Computer-Aided Diagnosis (CAD) for Breast Cancer Detection," has been reviewed in final form. Permission, as indicated by the signatures and dates below, is now granted to submit final copies to the College of Graduate Studies for approval.

Major Professors: _____ Date: _____

Min Xian, Ph.D.

Committee Members: _____ Date: _____

Aleksandar Vakanski, Ph.D.

_____ Date: _____

Xiaogang Ma, Ph.D.

Department

Administrator: _____ Date: _____

Terry Soule, Ph.D.

Abstract

Deep learning-based computer-aided diagnosis systems have achieved unprecedented performance in breast cancer detection. However, most approaches are computationally intensive, and impede their broad dissemination. In this study, we propose an efficient and light-weighted multitask learning framework to simultaneously classify and segment breast tumors. Pretrained MobileNetV1 is employed as the multitask network backbone, followed by 2 branches for classification and segmentation respectively. The segmentation branch utilizes the Link-Net as decoder. The proposed approaches are evaluated using a dataset with 864 B-scan breast ultrasound images. Extensive experiments demonstrate that the proposed multitask learning network not only improves the classification and segmentation accuracy, but also keeps the low latency and efficiency properties. The network achieves 86.6% Dice's coefficient and 79% Intersection Over Union for segmentation, while 93.85 % Accuracy, 94.44% Sensitivity and 93.42 Specificity for classification. The trained network has the size of around 60 MB in Keras H5 format, and 20 MB after converting to Tensorflow Lite format. Lastly, we develop and build a mobile application in Android Studio. It launches the trained multitask learning network to do real-time breast tumor detection, with average inference time cost for classification and segmentation together being around 300 milliseconds.

Acknowledgments

First and foremost, I would like to thank all members of my graduate committee.

I am extremely fortunate to meet with Dr. Xian in my first semester at the University of Idaho, and have him as my major professor. He is a great supervisor, has always been so inspiring, patient, and giving endless guidance, and constructive support to not only my course study, research but also career development. Without his mentoring, timely advice, and encouragement, I could not make such progresses in my graduate study and accomplish this thesis. It is a genuine pleasure to express my deep sense of gratitude to Dr. Xian.

Although it is my last semester when I met with Dr. Vakanski, I was very expressed by his kindness and dedication to students and expertise at Adversarial Machine Learning. I would like to sincerely appreciate Dr. Vakanski for all of his help, guidance and constructive advices through my research progress.

I have taken several classes from Dr. Ma, and I feel grateful that taking his Data Science course unlocked my career as a data scientist. Dr. Ma's humor in classes brings us a lot of fun with study while his professional knowledge in Geosciences and data analytics always introduce us to deeper insights and fresh understanding. I would like to thank Dr. Ma for all of his help and willingness of guidance with his abilities.

Dedication

Most importantly, I would like to thank my family and beloved one and wholeheartedly dedicate this thesis to them. Without their support, strength, and caring, I would have never had the chance to pursue my Master of Science degree at the University of Idaho.

Table of Contents

Authorization to Submit Thesis ii

Abstract iii

Acknowledgments iv

Dedication v

Table of Contents 1

List of Tables 3

List of Figures 4

CHAPTER 1: INTRODUCTION 6

1.1 Breast Cancer Detection and Computer-Aided Diagnosis (CAD) 6

1.2 Breast Cancer Classification Using Deep Learning 7

1.3 Breast Cancer Segmentation Using Deep Learning 7

1.4 Efficient Multitask Learning for BUS Classification and Segmentation 8

1.5 Thesis Organization 9

CHAPTER 2: EFFICIENT MULTITASK NETWORK FOR BREAST CANCER DETECTION 10

2.1 Efficient Breast Cancer Classification Using MobileNet 10

2.2 Efficient Breast Cancer Segmentation with LinkNet 12

2.3 MobileNet-LinkNet Based Multitask Learning for Breast Cancer Detection 14

CHAPTER 3: BREAST ULTRASOUND DATASET 16

3.1 Breast Ultrasound (BUS) Dataset 16

3.2 Data Preprocessing 20

CHAPTER 4: EXPERIMENTAL RESULTS 26

4.1 Experimental Settings and Evaluation Metrics 26

4.2 Single-task tumor Classification 30

4.2.1 Padding versus Cropping 30

4.2.2 Classification results with different pretrained efficient CNNs 32

4.2.3 Weighted Binary Cross-Entropy 35

4.2.4 Classification Using Image Augmentation 36

4.3 Multi-task Tumor Classification and Segmentation 39

4.3.1 Single-task Efficient Tumor Segmentation 39

4.3.2 Efficient Multitask Tumor Classification and Segmentation 41

4.4 Mobile Application Development Using Android Studio 43

CHAPTER 5: CONCLUSION 46

References 48

List of Tables

Table 1. BI-RADS Classification Category and Scoring..... 18

Table 2. Number of Benign and Malignant Tumors in datasets..... 19

Table 3. Number of Horizontal and Vertical Tumors in datasets..... 21

Table 4. K-fold ACC (%) of the padding and cropping strategy. 31

Table 5. Size and Parameters of the pretrained efficient CNNs..... 32

Table 6. Size and Parameters of the classification model with pretrained CNNs..... 33

Table 7. K-fold ACC (%) of the classification model with pretrained CNNs. 33

Table 8. Tumor classification using various augmentation techniques. 38

Table 9. Size and Number of Parameters of evaluated segmentation models. 40

Table 10. Performance of the 4 evaluated segmentation models. 40

Table 11. Segmentation and classification performance of proposed multitask network. 42

Table 12. Segmentation and classification performance of different weight ratios. 43

List of Figures

Figure 1. The architecture of the MobileNetV1. DS represents depthwise separable convolution, which is a sequential combination of a 3 x 3 depthwise convolution and a 1 x 1 pointwise convolution, each followed by Batchnormlization operation and Relu activation.. 11

Figure 2. LinkNet Architecture [42]. (a) Convolutional modules in encoder-block; (b) Convolutional modules in decoder-block; and (c) Overall Architecture. 13

Figure 3. MobileNet-LinkNet Architecture. (a) Overall architecture with MobileNetV1 as the backbone, whose last convolutional layer followed by the classification branch with new added dense layers as the binary classifier, and the segmentation branch using the LinkNet encoder; and (b) the internal connections between the MobileNetV1 LinkNet. 15

Figure 4. Example of a (a) Breast Anatomy, (b) Mammography, and (c) breast ultrasound imaging [45]. 17

Figure 5. Examples of 4 common abnormalities seen in the breast [45]: (a) Cysts with anechoic pattern, oval shape, circumscribed margin, horizontal orientation, posterior enhancement and no calcifications; (b) Fibroadenomas with hypoechoic pattern, oval shape, circumscribed margin, horizontal orientation, minimal posterior enhancement and gross calcifications; (c) Breast Cancer with hypoechoic pattern, irregular shape, uncircumscribed margin, vertical orientation, posterior shadowing and small calcifications; (d) Locally palpable glandular tissue with hyperechoic pattern and no calcifications. 17

Figure 6 Examples of benign (left) and malignant (right) tumor images, separately from BUSIS dataset (row a), Dataset B (row b), and Thailand dataset (row c). 20

Figure 7. An example of image whose tumor is large and close to boundaries (a). To avoid cutting the tumor instance to be incomplete by the cropping (b), use padding instead (c). 23

Figure 8. An example of an original horizontal shaped image (left) and its mask (right) and different resized results: (a) directly resized from the original; (b) resized after the padding; (c) resized after the cropping. 24

Figure 9. An example of an original vertical shaped image (left) and its mask (right) and different resized results: (a) directly resized from the original; (b) resized after the padding; (c) resized after the cropping. 25

Figure 10. Training ACC and Loss Curves of running two times using same data, model, and environment configurations, on GPU (a) and CPU (b), respectively. 27

Figure 11. K-fold ACCs of the tumor classification trained on images resized directly from original images, resized after padding, and resized after cropping. 31

Figure 12. Results of K-fold cross-validation of four efficient CNNs using two random seeds to split data in each fold..... 34

Figure 13. ACC, SEN and SPE of tumor classification using different weights in the WBCE loss function. 36

Figure 14. Example of (a) an original resized image with (b) width shifting by 10%, (c) height shifting by 10%, and (d) horizontal flipping. 38

Figure 15. K-fold ACCs of the tumor classification using different augmentation techniques. 39

Figure 16. Performance Comparison Results of different segmentation models..... 41

Figure 17. Examples of two original image (left), ground truth (middle), and segmentation results (right) from the multitask model. 42

Figure 18. Figure 18. Mobile Application workflow. (a) landing page; (b) mobile photo gallery; (c) a benign case and the results; and (d) a malignant case and the results. 44

CHAPTER 1: INTRODUCTION

1.1 Breast Cancer Detection and Computer-Aided Diagnosis (CAD)

Breast cancer is one of the most common cancers and leading causes of death among women worldwide [1] [2]. Alone in 2019 in the U.S., around 268,600 new cases of invasive breast cancer and 62,930 new cases of non-invasive breast cancer in women were expected to be diagnosed [3]. Early detection is the best way to reduce the mortality rate and improve treatment outcomes. Breast ultrasound (BUS) imaging has become one of the critical and effective imaging methods to detect and classify breast tumors because of its non-invasive, nonradioactive, and cost-effective nature and the easy integration into interventional procedures for future patient treatments [4]. Furthermore, BUS imaging is the most suitable tool for large-scale breast cancer screening and diagnosis in low-resource countries and regions. Over the decades, it has been demonstrated that BUS has several significant advantages over other medical imaging modalities such as X-ray, magnetic resonance imaging (MRI), and computed tomography (CT), including its non-ionizing radiation, portability, accessibility, and cost-effectiveness. However, in clinical routine, both automated BUS image classification and segmentation face unique challenges and heavily depend on the experience and technical ability of the clinical operators. Thus, the divergence in the operators' skills, knowledge, and understanding of various BUS techniques is highly likely to lead to observation and determination variations during the diagnosis practice. Computer-aided diagnosis (CAD) systems are developed [5] to improve the predictive accuracy and overcome the operator dependency. CAD systems assist the radiologists with image interpretation and diagnosis by providing a second objective opinion. Many CAD systems have been clinically tested and proved their ability to improve the diagnostic sensitivity, specificity, and efficiency of the breast cancer diagnosis [6].

1.2 Breast Cancer Classification Using Deep Learning

Significant progress in medical image processing has been achieved by using deep learning techniques in recent years. Applying deep learning with Convolutional Neural Network (CNN) to BUS image classification has gained popularity in clinical practice because it achieves unprecedented performance, saves time, reduces radiologist fatigue, and compensates for lack of experience and skills in some cases. Since AlexNet [7], a representative deep CNN architecture and winner of the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), deep learning began to attract attention. To date, many studies have reported the usefulness of deep learning for diagnostic imaging of breast masses with ultrasound, e.g., discrimination between benign and malignant breast mass images from ultrasound [8], and classification of breast lesions [9]. Shi *et al.* [10] used the stacked deep polynomial network to extract textural features from 100 malignant and 100 benign masses on ultrasound images [10]. Stoffel *et al.* [11] focused on phyllodes tumor and fibroadenoma classification using deep learning and achieved good accuracy (AUC = 0.73) and high negative predictive value (NPV = 100%). GoogLeNet [12] is used in [13], where 4254 benign samples and 3154 malignant samples were employed to train the deep CNN, and reported an accuracy of 91%, a sensitivity of 86%, a specificity of 93%, and an AUC over 0.9. Shin *et al.* [14] applied transfer learning to achieve better classification performance by pretraining a CNN using the ImageNet dataset and fine-tuning the network using a BUS dataset. Cheng *et al.* [15] used the CaffeNet [55] and VGG [28] with frozen pretrained weights from transfer learning for abdominal ultrasound images.

1.3 Breast Cancer Segmentation Using Deep Learning

BUS image segmentation aims to extract tumor regions from normal breast tissues in images, which is an essential step in the CAD systems for breast cancer detection. However, because of the speckle noise, poor image quality, and variable tumor shapes and sizes,

achieving accurate BUS image segmentation is challenging [16]. CNN-based segmentation methods have been quite popular and have been proven to be highly accurate [17]. Various state-of-the-art BUS segmentation methods have been studied extensively, and good performances have been reported utilizing their own datasets. There still are common and fundamental issues in BUS segmentation, including the denoising and preserving edges, operator-dependency, and the results being unreproducible caused by the user interactions, lack of modeling domain-related knowledge [18]. In addition, it is well known that manual segmentation to get the ground truth masks are time-consuming and tedious, also suffers from great individual variability.

CNN-based models, such as Fully Convolutional Network (FCN) [19], SegNet [20], and U-Net [21], have been developed and widely used. A Residual-Dilated-Attention-Gate-U-Net (RDAU-NET) is an improved version of the regular U-Net and achieved a precision of 88.58%, sensitivity of 83.19%, and F1 score of 0.848 [22]. By combining a dilated fully convolutional network with a phase-based active contour model for automatic tumor segmentation, the constructed model is reported to exhibit high robustness, accuracy, and efficiency [23]. Kumar *et al.* [24] introduced the multi-U-net algorithm for automatic and effective breast masse segmentation. They achieved a mean Dice coefficient of 82%, a true positive rate of 84%, and a false positive rate of 0.01. Xian *et al.* [25] proposed a fully automatic and adaptive ROI generation method, where the ROI seeds can be generated with high accuracy and can be used to distinguish the lesion regions from normal regions.

1.4 Efficient Multitask Learning for BUS Classification and Segmentation

This work investigates both classification and segmentation approaches' accuracy and efficiency and aims to build a mobile CAD system for breast cancer detection. State-of-art CNN models can achieve great performance but are faced with issues of the large number of parameters and high hardware requirements. It is difficult to deploy these approaches to

mobile devices [26]. In this thesis, a multitask learning network is proposed to combine the two separate tasks together in one shared model to obtain both the tumor classification and segmentation results simultaneously; and the new approach can avoid deploying two independent models to model devices. In addition, the tradeoff between accuracy and efficiency is studied in this work. The study demonstrates that using multitask learning to combine the efficient classification and segmentation network as a single network could help boost the classification task while keeping the efficiency.

1.5 Thesis Organization

The thesis is organized as follows. Chapter 2 introduces several efficient classification models and segmentation models, and provides an overview of the proposed multitask model for the mobile CAD. Chapter 3 describes the BUS image datasets from three data sources used for training, validation, and testing approaches. The experimental results of single-task classification, single-task segmentation, and multitask modeling are introduced in Chapter 4. Chapter 4 also presents the final mobile application design using Android Studio. Chapter 5 briefly summarizes the work and concludes the thesis.

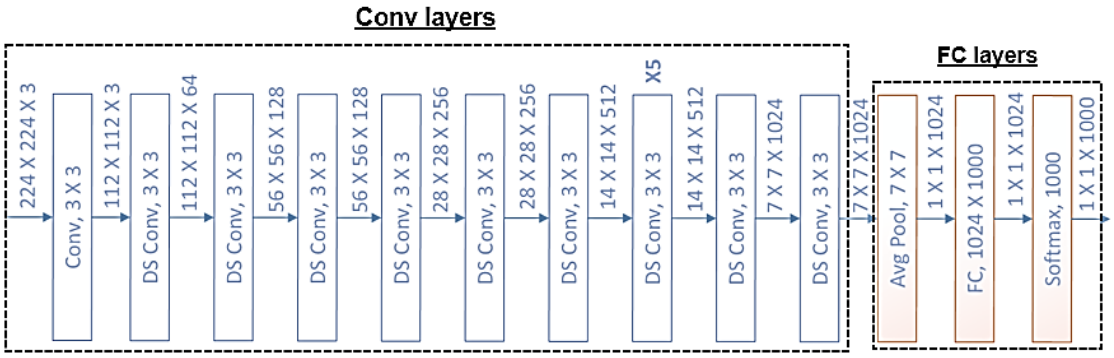
CHAPTER 2: EFFICIENT MULTITASK NETWORK FOR BREAST CANCER DETECTION

2.1 Efficient Breast Cancer Classification Using MobileNet

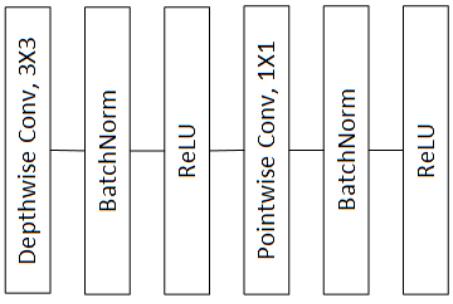
Deep CNNs have become ubiquitous and popularized in computer vision tasks. One of the general trends is to build deeper and more complicated architectures for higher performance to deal with more complex scenarios in real life. Many CNN architectures were proposed in the last decade, e.g., AlexNet [7], VGG [28], ResNet [29], DenseNet [30] and SeNet [31]. However, large number of parameters prevent the models to be deployed as real world application and make it less accessible to large group of users. Smart wearable devices with limited computational resources are getting populated, and real-time applications on mobile devices is witnessed with significant progress and rising popularity [32]. Efficient CNN infrastructure is in unprecedented demand.

Many small CNNs have been developed. Flattened networks [33] builds a network out of fully factorized convolutions out of the purpose of fast feedforward execution. Factorized Networks [34] introduces the use of topological connections. Squeeze-Net [35] uses a bottleneck approach to obtain a very small network. MobileNetV1 [36] primarily employs the primarily separable convolutions to substantially improve computation efficiency. MobileNetV2 [37] introduces a resource-efficient block with inverted residuals as well as linear bottlenecks which could help make even more efficient layers through leveraging the low rank nature of the problem. NASNet [38] searches for an architectural building block on a small dataset and then transfers the block to a larger dataset with a new regularization technique called ScheduledDropPath which significantly improves the generalization. Efficientnet [39] uses a simple while highly effective compound coefficient to scale up CNN to be in a more structured manner.

MobileNetV1 could be applied to various image recognition tasks for efficient on device intelligence, including Fine Grained Recognition, Large Scale Geo-localization, face attribute classification task and object detection. MobileNetV1 expresses the standard convolution with the depthwise separable convolution [40], which is a combination with 3 × 3 depthwise and 1 × 1 pointwise convolution layers. The depthwise convolutions perform lightweight filtering by applying a single convolutional filter per input channel. The 1 × 1 pointwise convolution layers build new features through computing linear combinations of the input channels, which is followed by Batchnorm [41] and Relu for robustness when used with low-precision computation. The depthwise separable convolution brings the benefit of only consuming around 8 to 9 times less computations than the standard convolution with only a small reduction in accuracy. Figure 1 shows the architecture of the MobileNetV1, including the structure of one single depthwise separable convolution.



(a) MobileNetV1 architecture



(b) DS Block

Figure 1. The architecture of the MobileNetV1. DS represents depthwise separable convolution, which is a sequential combination of a 3 x 3 depthwise convolution and a 1 x 1 pointwise convolution, each followed by Batchnormlization operation and Relu activation.

MobileNetV2 is built upon the ideas from MobileNetV1 using depth wise separable convolution as efficient building blocks. To overcome the risk of collapsing the channel which inevitably loses information in that channel caused by Relu in MobileNetV1, the V2 introduces two new features to the architecture, including the linear bottlenecks between layers, and the short connections between bottlenecks. In Efficientnet, a new scaling method which is designed to uniformly scale all dimensions of depth/width/resolution using a simple yet highly effective compound coefficient.

2.2 Efficient Breast Cancer Segmentation with LinkNet

Most of the popular CNNs-based segmentation models use the similar encoder-decoder architecture. Although initially developed for medical/biomedical image segmentation, U-Net could achieve state-of-art results of segmenting biological microscopy images. It is not designed to be efficient enough to deploy in real-time applications in terms of parameters and number of operations, hence it is quite slow too. Two parts are comprised in the U-Net architecture, including a contracting path to capture context and a symmetric expanding path that enables precise localization.

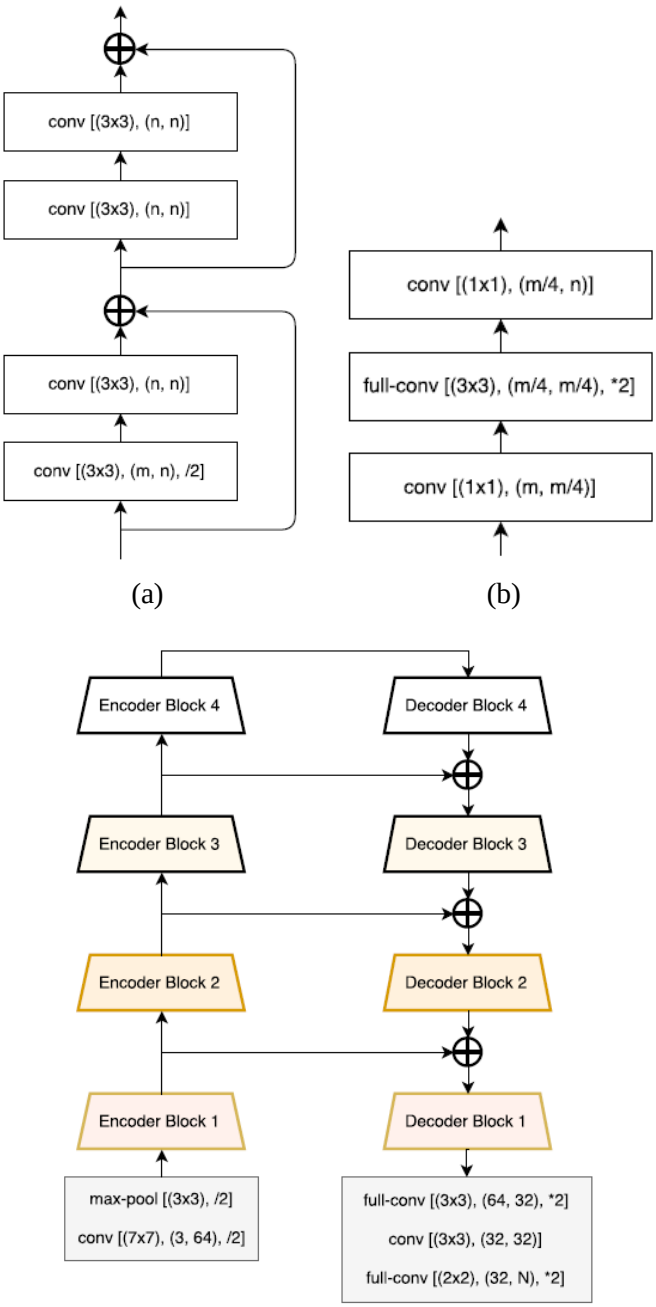


Figure 2. LinkNet Architecture [42]. (a) Convolutional modules in encoder-block; (b) Convolutional modules in decoder-block; and (c) Overall Architecture.

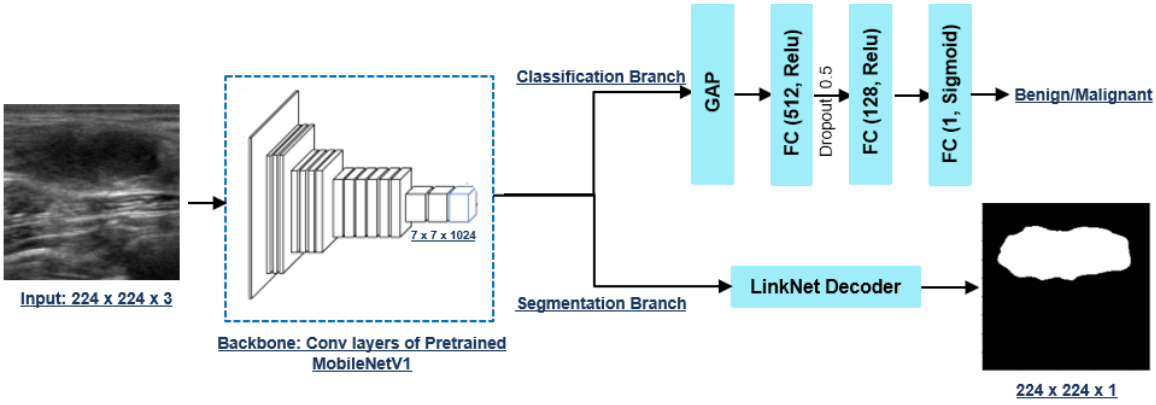
LinkNet [42] is a light weighted deep neural network architecture designed for performing semantic segmentation, and capable of giving real-time performance for tasks such as self-driving cars. It takes the advantages of skip connections, residual blocks and encoder-decoder architecture, and originally uses the ResNet18 [43] as its encoder, while in our

experiments, the encoder part will be replaced with the pretrained more efficient MobileNetV1. The original LinkNet structure is shown in Figure 2, where down-sampling by a factor of 2 is achieved by performing the stride convolution; the encoder part consists of residual blocks; and the novelty comes from the way each encoder being linked with decoder to recover the potential lost spatial information caused by multiple down-sampling operations.

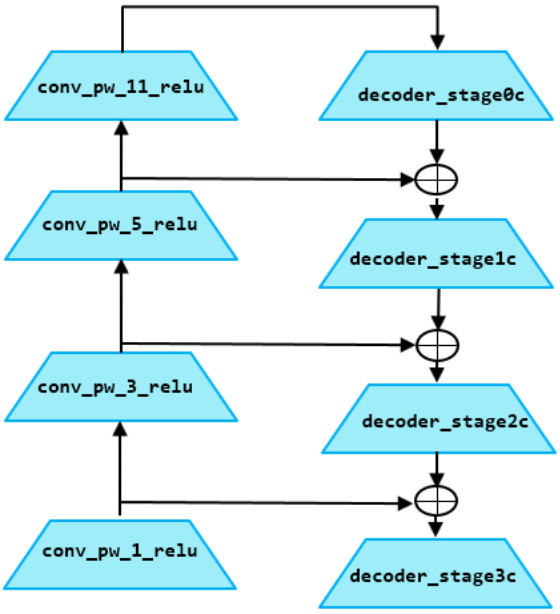
2.3 MobileNet-LinkNet Based Multitask Learning for Breast Cancer Detection

As mentioned in the Introduction, deploying two separate deep learning models will result in two large models running on mobile devices. This section studies automatically and jointly achieve BUS image segmentation and classification. The segmented lesions could help assess the tumor severity if diagnosis as malignant and assist further follow-ups. A multi task learning architecture with both classification and segmentation tasks is proposed in this section. The motivation is to leverage useful information from two related tasks to improve performances of each other [44].

The pretrained MobileNetV1 is used as the shared backbone network and encoder part to extract the local features representation from the input images efficiently. Figure 3 shows both the overall model with classification and segmentation branches, along with the detail for layers in encoder linking with layers in decoder. In the pretrained MobileNetV1, only the convolutional layers (Figure 1) are used with the output shape being 7 x 7 x 2024, while its original FC layers are not included.



(a)



(b)

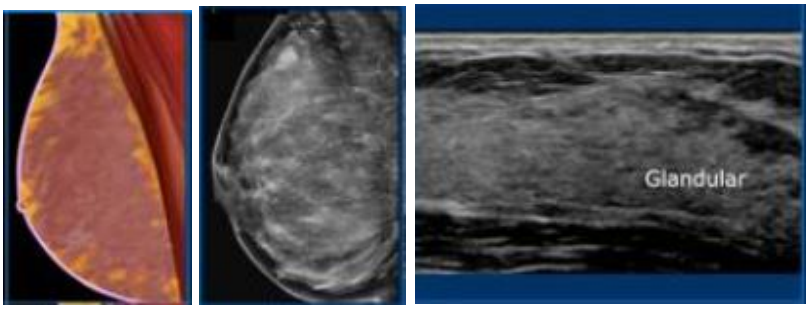
Figure 3. MobileNet-LinkNet Architecture. (a) Overall architecture with MobileNetV1 as the backbone, whose last convolutional layer followed by the classification branch with new added dense layers as the binary classifier, and the segmentation branch using the LinkNet encoder; and (b) the internal connections between the MobileNetV1 LinkNet.

CHAPTER 3: BREAST ULTRASOUND DATASET

3.1 Breast Ultrasound (BUS) Dataset

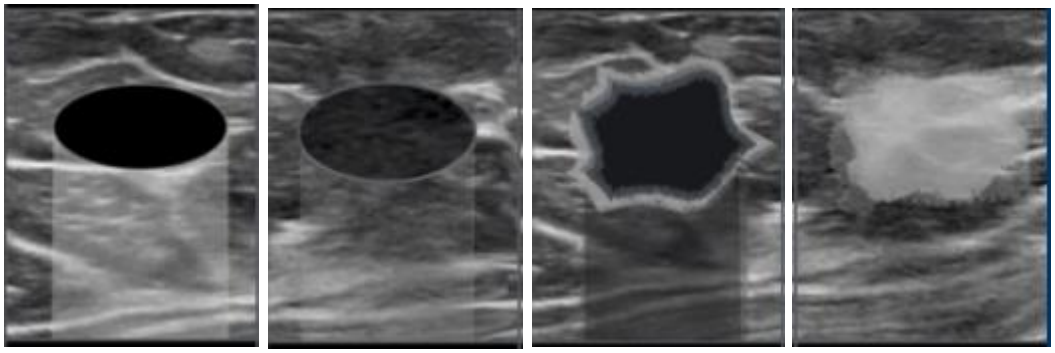
The introduced approaches are evaluated using BUS datasets. Sound waves are used in BUS instead of radiation or x-ray and sent out by moving a wand-like instrument called a transducer screening tools across over the skin. BUS generates a computer picture of the interior of the breast, where certain breast changes could be easily shown, especially for those changes like fluid-filled cysts, that could be perceived by human but are relatively difficult to be identified on mammograms. In addition, echoes that bounce off the body tissues could be also picked up into the picture, as echo patterns and echogenicity is used as one criterion in diagnosis for benign or malignant lesions. Figure 4 shows an example of mammograph and BUS imaging for same Breast Anatomy [45].

In real clinical practice of diagnosis on breast screening ultrasounds, image characteristics including tumor shape, orientation, margin, echo pattern, posterior features, and associated calcifications features are all important assessment features, the details of standardized terminology to describe Breast Ultrasound findings using BI-RADS (Breast Imaging-Reporting and Data System) could be found in [46]. Figure 5 shows examples for 4 common abnormalities represented as a lump in BUS images along with corresponding ultrasound findings.



(a) (b) (c)

Figure 4. Example of a (a) Breast Anatomy, (b) Mammography, and (c) breast ultrasound imaging [45].



(a) (b) (c) (d)

Figure 5. Examples of 4 common abnormalities seen in the breast [45]: (a) Cysts with anechoic pattern, oval shape, circumscribed margin, horizontal orientation, posterior enhancement and no calcifications; (b) Fibroadenomas with hypoechoic pattern, oval shape, circumscribed margin, horizontal orientation, minimal posterior enhancement and gross calcifications; (c) Breast Cancer with hypoechoic pattern, irregular shape, uncircumscribed margin, vertical orientation, posterior shadowing and small calcifications; (d) Locally palpable glandular tissue with hyperechoic pattern and no calcifications.

Using BUS criteria, a lesion can be assigned into one of seven BI-RADS categories using a scoring system developed by the ACR with the following interpretation and management shown in Table 1. Each BIRADS category corresponds to a classification that estimates the breast health and cancer risk obtained from the diagnosis.

Table 1. BI-RADS Classification Category and Scoring

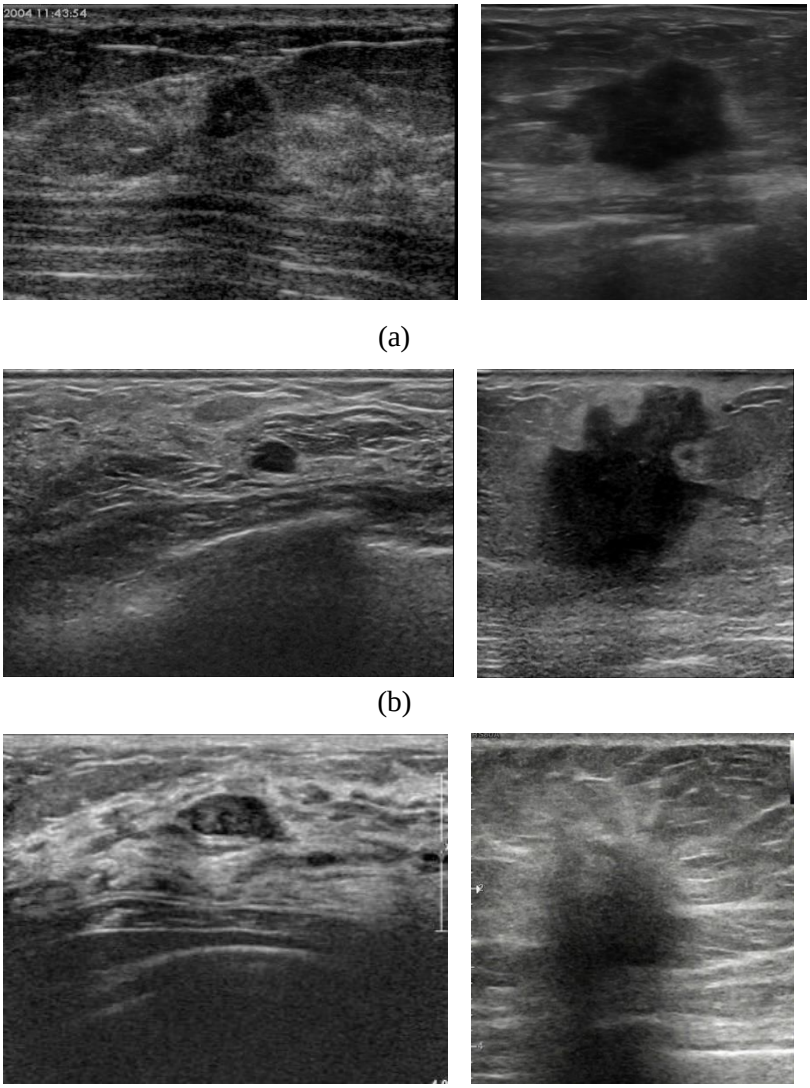
BIRADS Category	Diagnosis Description & Management
0	Incomplete. Need additional imaging evaluation.
1	Negative. Symmetrical and no masses, architectural distortion or suspicious calcifications. Routine screening recommended.
2	Benign. 0 % probability of malignancy. Early follow-up to document stability unless clinical indications suggest a more aggressive evaluation.
3	Probably benign. Probability of malignancy <2%. Short interval follow-up suggested.
4	Suspicious malignancy. Not characteristic of breast cancer, but reasonable probability of being malignant. Further divided: • 4A: low suspicion for malignancy (2-9%) • 4B: moderate suspicion for malignancy (10-49%) • 4C: high suspicion for malignancy (50-94%) Biopsy should be considered.
5	Highly suggestive of malignancy. >95% probability of malignancy. Appropriate action should be taken.
6	Known biopsy-proven malignancy. Lesions known to be malignant that are being imaged prior to definitive treatment. Assure that treatment is completed.

The entire dataset is consisted of three public Breast Ultrasound datasets: the BUSIS dataset [47], the Dataset B [48] and the Thailand dataset [49]. Patients' privacy is well guaranteed and protected. The BUSIS benchmark dataset contains total 562 images collected by the Second Affiliated Hospital of Harbin Medical University, the Affiliated Hospital of Qingdao University, and the Second Hospital of Hebei Medical University using different ultrasound devices including of GE VIVID 7, LOGIQ E9, Hitachi EUB-6500, Philips iU22, and Siemens ACUSON S2000. The Dataset B consists of total 163 lesion images with a mean image size of 760 by 570 number of pixels, which are collected in 2012 by from the UDIAT Diagnostic Centre of the Parc Tauli Corporation, Sabadell (Spain) with a Siemens ACUSON Sequoia C512 system 17L5 HD linear array transducer (8.5 MHz). There are one

or more than one lesions presented in each of the images. The Thailand dataset has total 139 lesion images, collected from the US Online Medical Images database provided by the Department of Radiology of Thammasat University and Queen Sirikit Center of Breast Cancer of Thailand. Images along with the ground truth are hand-drawn by leading radiologists of these centers. Table 2 gives the number of benign lesions and malignant lesions for each of the three datasets separately. Figure 6 shows examples of both benign and malignant lesion images in BUSIS dataset, Dataset B and Thailand dataset. Combining the three datasets could help to enhance and investigate the robustness of training and evaluating the introduced models.

Table 2. Number of Benign and Malignant Tumors in datasets

Dataset	**Images**	**Benign**	**Malignant**	**Benign ratio**	**Malignant ratio**
BUSIS	562	306	256	54.4%	45.6%
Dataset B	163	109	54	66.9%	33.1%
Thailand	139	52	87	37.4%	62.6%
Entire dataset	**864**	**467**	**397**	**54.1%**	**45.9%**



(a)

(b)

(c)

Figure 6. Examples of benign (left) and malignant (right) tumor images, separately from BUSIS dataset (row a), Dataset B (row b), and Thailand dataset (row c).

3.2 Data Preprocessing

Almost all the images in the datasets are originally rectangle shaped, however all the introduced classification and segmentation models require inputs with fixed and equal width and height. From the perspective of real clinical practice, clinical doctors rely a lot on the tumor shape so that with reasonable descriptions for benign and malignancy diagnosis and determination. From the perspective of modeling, original tumor shape in the image is also an important feature that the model leverages for better training and correct identification. In order to have the original tumor shape kept unchanged while the image shape processed

from rectangle to square, obviously simply implementing resize operation over the whole image will bring out the tumor distortion problem, as a result simple resize or rescale is not allows to be applied in our application. Table 3 shows the number and ratio of horizontal shape images and vertical shape images over the 3 datasets.

Table 3. Number of Horizontal and Vertical Tumors in datasets

Dataset	Images	Horizontal	Vertical	Horizontal Ratio	Vertical Ratio
BUSIS	562	490	72	87.2%	12.8%
Dataset B	163	133	33	81.6%	18.4%
Thailand	139	104	35	74.8%	25.5%
Entire dataset	**864**	**727**	**137**	**84.1%**	**15.9%**

To avoid tumor distortion, two strategies are introduced as data preprocessing stage, including Padding strategy and Cropping strategy. Both the Padding and Cropping strategy need firstly leverage the tumor object from the ground truth for each image, to find the four vertex and get the four distance of each vertex to boundary.

In addition, considering the upper gray skin layer along with dark or hypoechoic fat layer and glandular tissue are useful information for accurate diagnosis and better classification, both the Padding and Cropping strategy will leave the upper layers as origin without adding new pixels or removing any pixels. Let $\boldsymbol{W}_{ori}$ denote the original width, $\boldsymbol{H}_{ori}$ denotes the original height, $\boldsymbol{S}_{diff}$ denote the absolute value of the difference between the $\boldsymbol{W}_{ori}$ and $\boldsymbol{H}_{ori}$, and $\boldsymbol{S}_t$ denote the target final squared size. Let $\boldsymbol{D}_l$, $\boldsymbol{D}_r$, $\boldsymbol{D}_b$ represent the distance from the left, right and bottom vertex to the boundary of corresponding direction, while $\boldsymbol{R}_l$, $\boldsymbol{R}_r$ represent the ratio of $\boldsymbol{D}_l$ and $\boldsymbol{D}_r$over the original image width.

$$\boldsymbol{R}_l = \frac{\boldsymbol{D}_l}{\boldsymbol{W}_{ori}}; \quad \boldsymbol{R}_r = \frac{\boldsymbol{D}_r}{\boldsymbol{W}_{ori}} \tag{1}$$

Padding strategy introduces new black pixels around the shorted edges of the original images and masks. For images being horizontal shaped, $\boldsymbol{S}_t$ will equal to $\boldsymbol{W}_{ori}$ and all new pixels will be padded at the bottom layers. For images being vertical shaped, $\boldsymbol{S}_t$ will equal

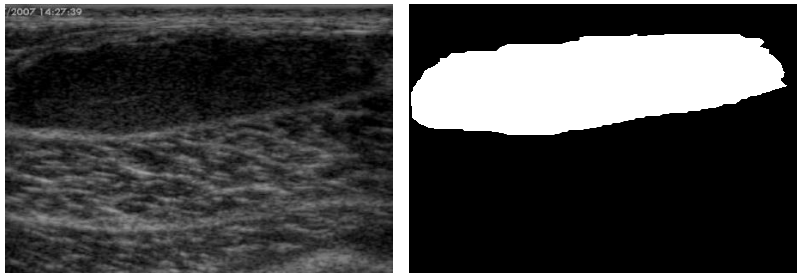
to $\boldsymbol{H_{ori}}$, new pixels will be added into both the left side (denoted as $\boldsymbol{P_l}$) and right side (denoted as $\boldsymbol{P_r}$) of images based on the $\boldsymbol{R_l}$ and $\boldsymbol{R_r}$. The number of pixels to be added at each side is calculated as

$$\boldsymbol{P_l = S_{diff} * R_l;\ P_r = S_{diff} - P_l} \quad (2)$$

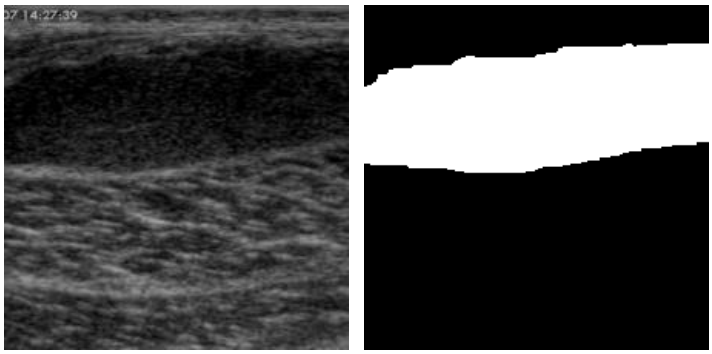
On the contrary with padding, the Cropping strategy removes pixels from the longer edges of the images and masks. For images being vertical shaped, $\boldsymbol{S_t}$ will equal to $\boldsymbol{W_{ori}}$ and all pixels of $\boldsymbol{S_{diff}}$ will be removed from the bottom layers. For images being horizontal shaped, $\boldsymbol{S_t}$ will equal to $\boldsymbol{H_{ori}}$ and pixels will be removed from both the left side (denoted as $\boldsymbol{C_l}$) and right side (denoted as $\boldsymbol{C_r}$) of images based on the $\boldsymbol{R_l}$ and $\boldsymbol{R_r}$. The number of pixels to be removed at each side is calculated as

$$\boldsymbol{C_l = S_{diff} * R_l;\ C_r = S_{diff} - C_l} \quad (3)$$

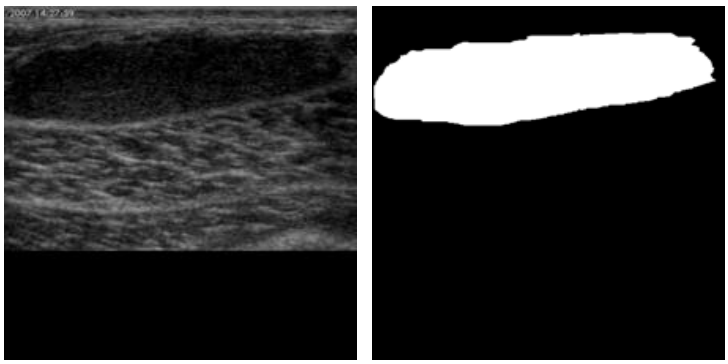
During the cropping preprocessing, there are cases observed whose tumor instance is very large and close to its left and right boundary, and makes the available space for cropping which calculated as the summation of $\boldsymbol{D_l}$ and $\boldsymbol{D_r}$ is smaller than $\boldsymbol{S_{diff}}$, as a result the tumor object inside the image will be cropped to be incomplete. For those images, the padding strategy is then applied as the supplement. See examples of this case in Figure 7.



(a)



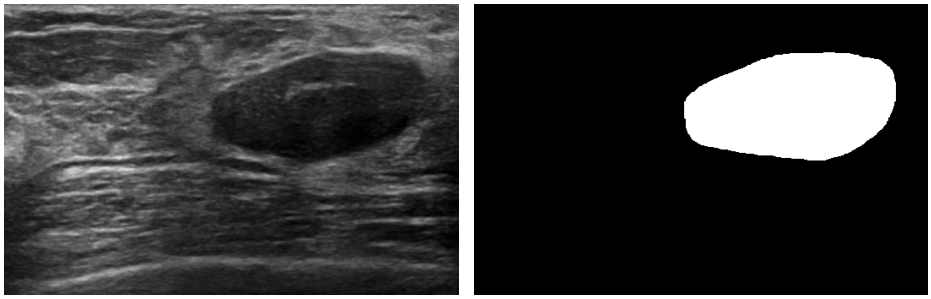
(b)



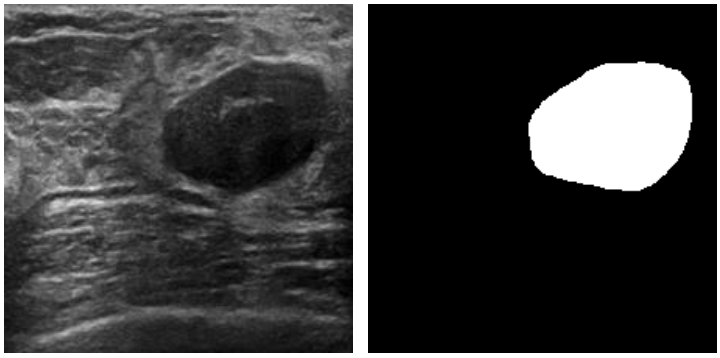
(c)

Figure 7. An example of image whose tumor is large and close to boundaries (a). To avoid cutting the tumor instance to be incomplete by the cropping (b), use padding instead (c).

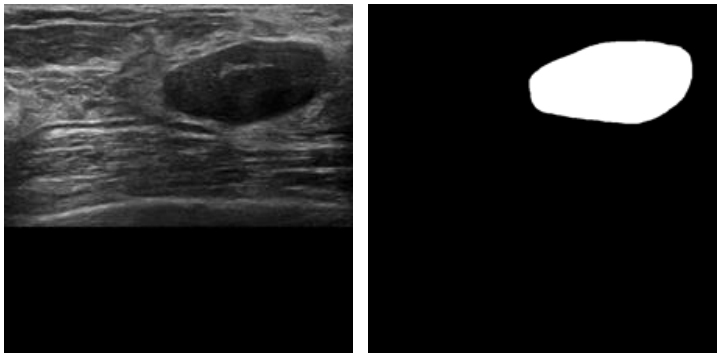
Examples given in Figures 8 and 9 are clearly shown the resized result from original image has caused severe tumor distortion, while Padding and Cropping avoid the distortion. Also, although both the Padding and Cropping prevents the tumor from transforming, padding will have the tumor taken smaller proportion over the entire image, compared with cropping, whose impact on the modeling accuracy will be introduced in the next experiment session.



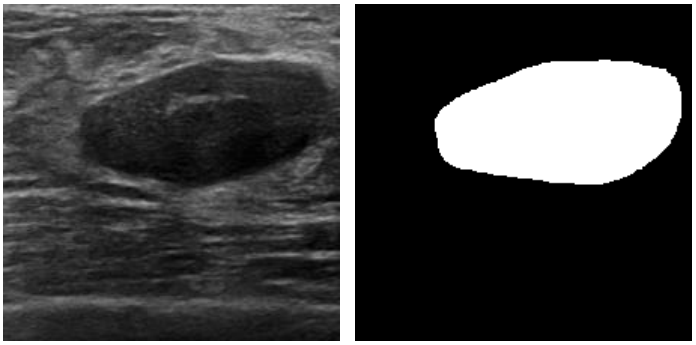
(a)



(b)

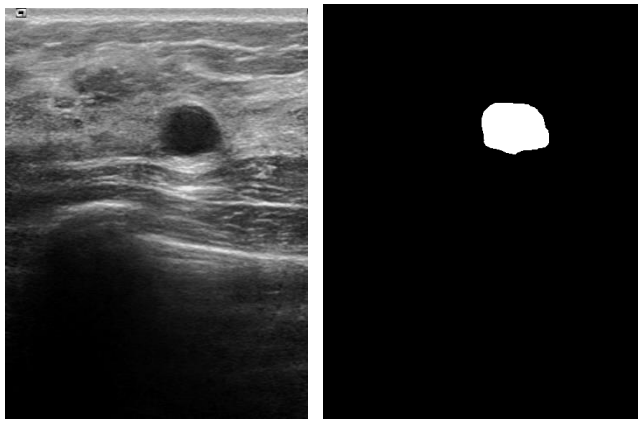


(c)

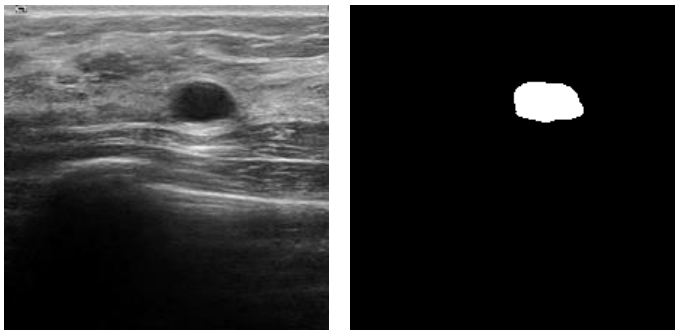


(d)

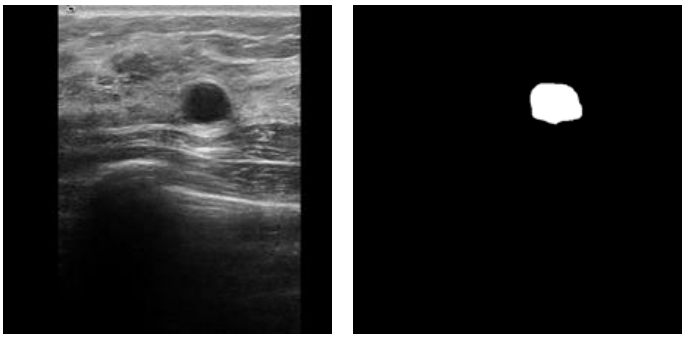
Figure 8. An example of an original horizontal shaped image (left) and its mask (right) and different resized results: (a) directly resized from the original; (b) resized after the padding; (c) resized after the cropping.



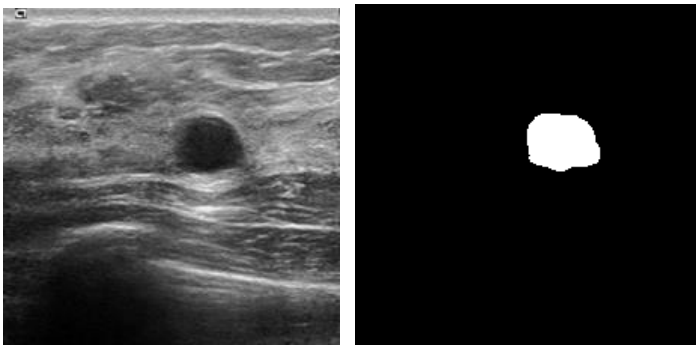
(a)



(b)



(c)



(d)

Figure 9. An example of an original vertical shaped image (left) and its mask (right) and different resized results: (a) directly resized from the original; (b) resized after the padding; (c) resized after the cropping.

CHAPTER 4: EXPERIMENTAL RESULTS

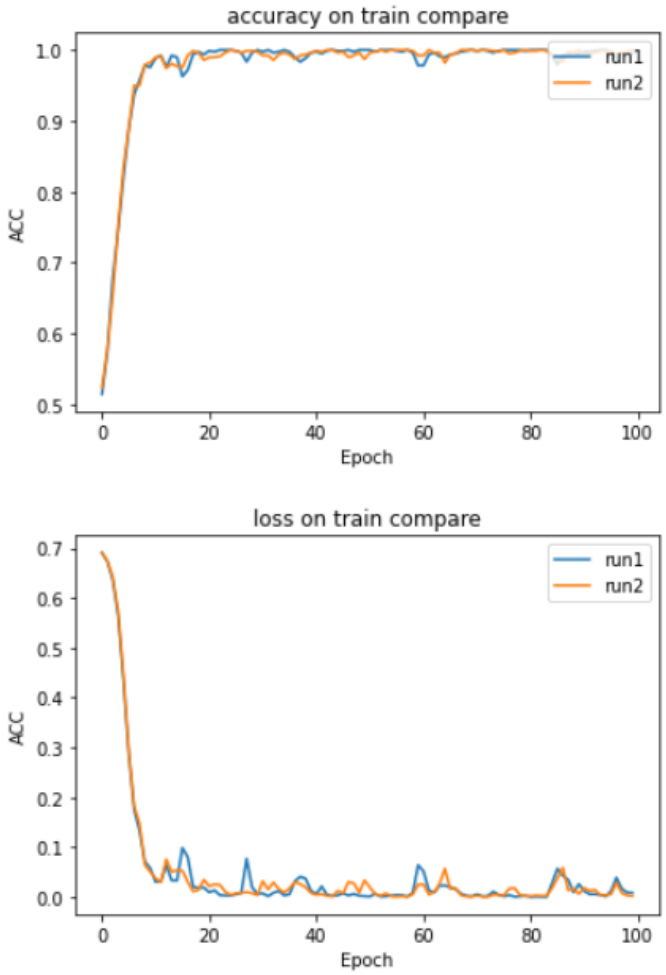
4.1 Experimental Settings and Evaluation Metrics

The experiments for classification and segmentation were performed using NVIDIA QUADRO RTX 8000 GPU, equipped with CUDA (version 10.2) and cuDNN (version 7). The neural network models are constructed in Keras (version 2.3.1) using Tensorflow (1.14.0 GPU version) as the backend. The input images and their corresponding ground truths are resized to a fixed size of 224 × 224. The percentages of train, validation, and testing sets used in the experiments are 70%, 15%, and 15%, respectively. In the K-fold cross-validation, K is set to 4 to have 75% of data into training while 12.5% in validation set and 12.5% in test set.

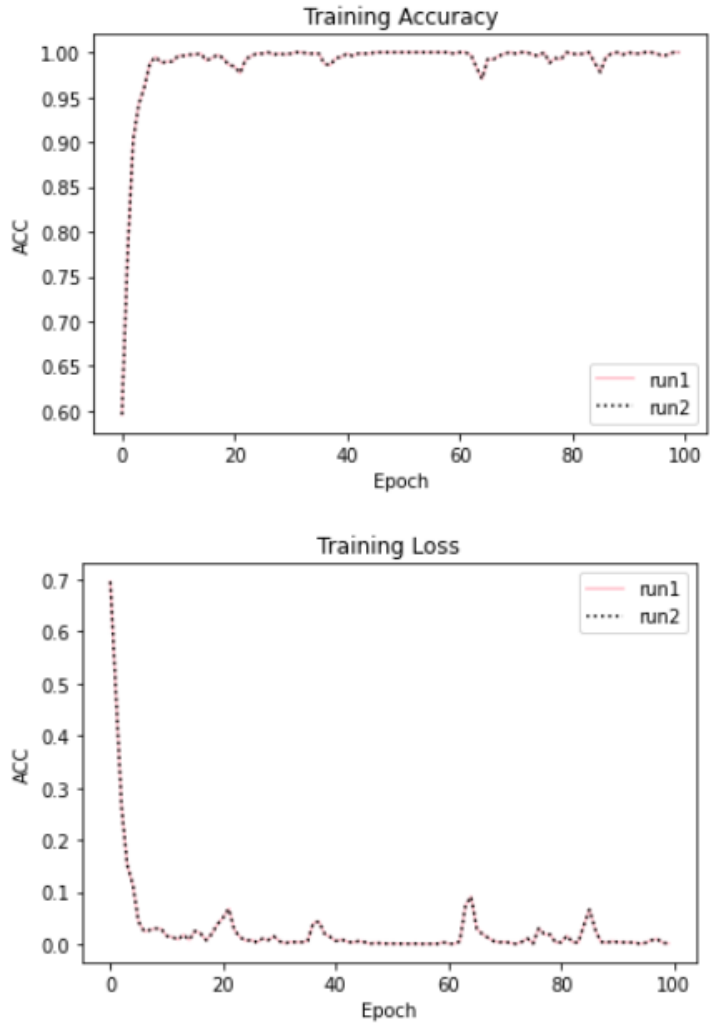
For the single classification task, transfer learning is used since the number of the entire training images is limited. Layers (not including the previous final dense layer) with weights pretrained using ImageNet are taken as the convolutional layers of the classifier. All the pretrained layers are set to be trainable for fine-tuning purposes and meaningful improvements by incrementally adapting the pretrained features to the new medical images. Following the pretrained convolutional layers, one Global Average Pool (GAP) layer [50], two new fully connected (FC) layers using Relu activation, and one last FC with Sigmoid activation are added. A dropout layer [51] with a dropout rate 0.5 is added after the first new sense layer. The batch size is 32, and the number of training epoch is set as 100. Stochastic gradient descent (SGD) optimizer is used with a learning rate 0.003 and a momentum (set to 0.9). The momentum [52] is used to accelerate the SGD in the relevant direction and dampens oscillations.

The randomness in many machine learning algorithms result in poor reproducibility. When comparing and demonstrating the impact of a specific approach, the randomness should be controlled. For example, comparing two image preprocessing strategies or different image augmentation techniques, and environment configurations, only reproducible results are meaningful for comparisons.

GPUs speed up the training at the expense of losing the reproducibility. In addition, in the Tensorflow, some common operations, e.g., reduce_sum, are by default designed to be non-deterministic on the GPU to make use of the CUDA atomics when broadcasting the biases; and Tensorflow is unable to control the randomness of its multithreading processing, which make it unlikely to achieve repeatability on the GPU.



(a) GPU results of two training rounds



(b) CPU results of two training rounds

Figure 10. Training ACC and Loss Curves of running two times using same data, model, and environment configurations, on GPU (a) and CPU (b), respectively.

As shown in Figure 10 (a), the two training rounds can reproduce well at early epochs (<10); but when the epochs keep increasing, differences between curves emerge, which demonstrates that the training on GPUs cannot be reproduced well for large epochs. Figure 10(b) shows that running on CPUs can achieve perfect reproducibility.

The strategies and settings to achieve high reproducibility on CPUs are listed below. Firstly, environment variables including the PYTHONHASHSEED, CUDA_VISIBLE_DEVICES (need to set to -1, if the Tensorflow version installed is a GPU version),

TF_DETERMINISTIC_OPS and TF_CUDNN_DETERMINISTIC are required to set to a fixed value, and in our experiments are all set to be 1. Secondly, the random seed should be set to a fixed value. Thirdly, to avoid any non-determinism from inter-operation parallelism, configurations in the Tensorflow for all inter- and intra-operation parallelism should be set to 1. Fourthly, when generating the train, validation, and test sets, the random state value is should be set to a fixed value (42 in experiments) so that each algorithm will be trained and evaluated using the same data. The seed value is set to 1 for the batch data generation for each training epoch. Lastly, the seed value of both the kernel weight initializer which uses the glorot uniform [53] should be fixed. For the single segmentation mask, the batch size is 32, and the optimizer uses Adam with an initial learning rate of 0.0001. To avoid learning stagnates, ReduceLROnPlateau function with 0.000001 as the minimum learning rate is added in the training callbacks.

Metrics for evaluating the classification include the accuracy (ACC), the sensitivity (SEN, also known as the true positive ratio (TPR)), and the specificity (SP, also known as the true negative ratio (TNR)). Metrics for evaluating the segmentation include the Dice's coefficient (DSC) [54], Intersection Over Union (IOU), TPR, TNR, ACC, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC).

$$\mathrm{ACC} = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})} \tag{4}$$

$$\mathrm{SEN} = \mathrm{TPR} = \frac{\mathrm{TP}}{(\mathrm{TP} + \mathrm{FN})} \tag{5}$$

$$\mathrm{SPC} = \mathrm{TNR} = \frac{\mathrm{TN}}{(\mathrm{FP} + \mathrm{TN})} \tag{6}$$

$$\mathrm{DSC} = \frac{2\mathrm{TP}}{2 \times \mathrm{TP}+\mathrm{FP}+\mathrm{FN}} \tag{7}$$

$$\mathrm{IOU} = \frac{\mathrm{TP}}{\mathrm{TP+FP+FN}} \tag{8}$$

where TP denotes the number of True Positives, TN the number of True Negatives, FP the number of False Positives, and FN the number of False Negatives.

4.2 Single-task tumor Classification

4.2.1 Padding versus Cropping

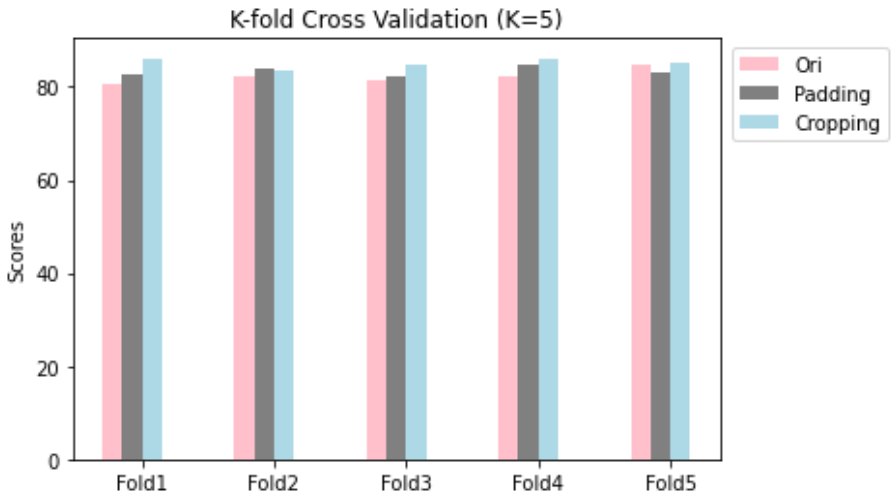
Two image preprocessing strategies are introduced to transform the image and corresponding ground truth from original rectangular shape to square shape without tumor distortion. One is applying padding to all images. The other is applying cropping to most images while using padding on those images that do not have enough space to be cropped because of large tumors.

Controlled K-fold (K = 5) cross-validation experiments using CPU are performed twice using two different random splitting seeds on (1) images resized directly from the original images, (2) images resized from the padding strategy, and (3) images resized from the cropping strategy. The accuracy of each fold and average accuracies are calculated and shown in Table 4 and visualized in Figure 11. As shown in Table 4, the cropping strategy outperforms the padding strategy averagely using two seeds for dataset splitting. All the experiments are performed on the resized images using the cropping preprocessing method in the later sections.

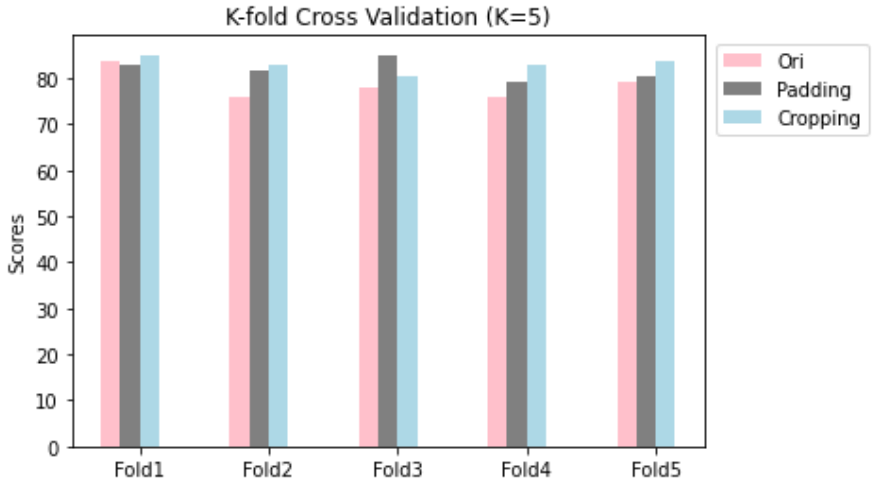
Table 4. K-fold ACC (%) of the padding and cropping strategy

		Original	Padding	Cropping
Seed = 42	Fold1	80.77	82.57	**86.15**
	Fold2	82.31	**84.85**	83.69
	Fold3	81.54	82.31	**84.62**
	Fold4	82.31	84.62	**86.15**
	Fold5	84.62	83.08	**85.38**
	Average	82.31	83.49	**85.20**
Seed = 77	Fold1	83.91	82.76	**85.06**
	Fold2	75.86	81.61	**82.76**
	Fold3	78.16	**85.06**	80.46
	Fold4	75.8	79.31	**82.76**
	Fold5	79.07	80.23	**83.72**
	Average	78.56	81.80	**82.95**

* The best accuracy in each row is highlighted using bold font.



(a) Seed=42



(b) Seed = 77

Figure 11. K-fold ACCs of the tumor classification trained on images resized directly from original images, resized after padding, and resized after cropping.

4.2.2 Classification results with different pretrained efficient CNNs

Transfer learning from pretrained models with loaded trainable weights is used as the base convolutional layers in the classification. The final CNN is then created from the base model, followed by two new dense layers with 512 and 128 units and Relu activation. A dropout layer with 0.5 dropout rate is inserted between these two dense layers to alleviate the risk of overfitting. The final dense layer is built with one unit and the sigmoid activation. Four efficient CNNs are compared to construct the new tumor classification network. Table 5 shows the model size and number of parameters for MobileNetV1, MobileNetV2, NASNetMobile, and EfficientNetB0. The Top-1 and Top-5 accuracies refer to the model's performance evaluated on validation set of the ImageNet dataset.

Table 5. Model size and parameters of the pretrained efficient CNNs.

Model	Size	Parameters	Top-1 Accuracy	Top-5 Accuracy
MobileNetV1	16 MB	4,253,864	0.704	0.895
MobileNetV2	14 MB	3,538,984	0.713	0.901
NASNetMobile	23 MB	5,326,716	0.744	0.919
EfficientNetB0	29 MB	5,330,571	0.767	0.931

Controlled K-fold experiments are performed with two random seeds to compare the performance of the four pretrained CNNs. All configurations are same, including training, validation, and testing splits, seed for weight initializers and seed the dropout layer. The maximum number of training epoch is 100, and the optimizer is SGD. All the experiments are conducted using CPU. Table 6 shows the model size and parameters of the new efficient CNNs.

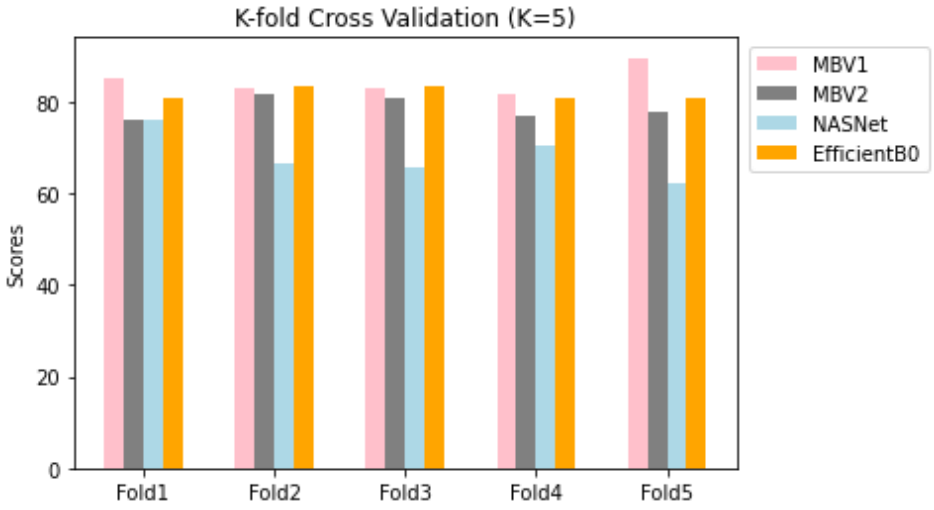
Table 6. Size and Parameters of the classification model with pretrained CNNs.

Pretrained Model	Size	Parameters
MobileNetV1	30 MB	3,819,457
MobileNetV2	24 MB	2,979,649
NASNetMobile	41 MB	4,876,693
EfficientNetB0	37 MB	4,771,229

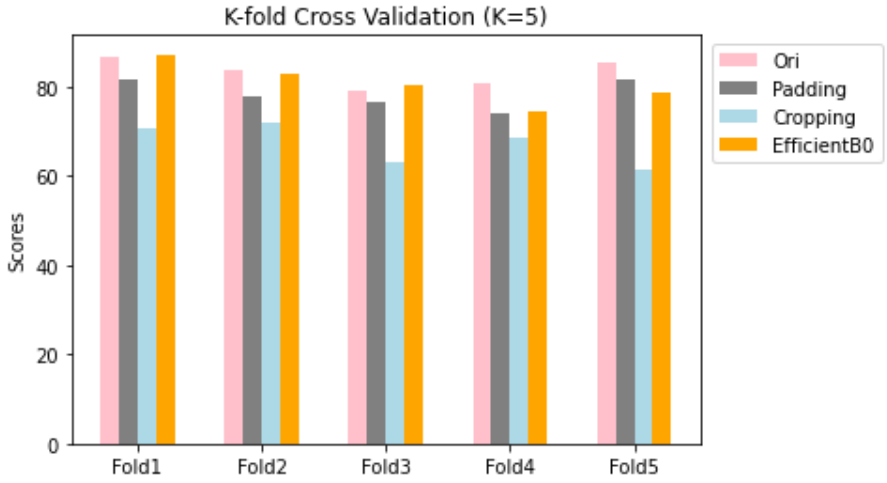
Table 7. K-fold ACC (%) of the classification model with pretrained CNNs

		MobileNetV1	MobileNetV2	NASNetMobile	EfficientNetB0
Seed = 42	Fold1	**85.06**	75.86	75.93	80.56
	Fold2	82.76	81.61	66.67	**83.33**
	Fold3	82.76	80.76	65.74	**83.33**
	Fold4	**81.46**	77.01	70.37	80.56
	Fold5	**84.62**	77.91	62.04	80.56
	Average	**84.314**	78.63	68.15	81.668
Seed = 77	Fold1	86.61	81.72	70.55	**87.23**
	Fold2	**83.61**	77.91	71.92	82.98
	Fold3	79.01	76.74	63.2	**80.5**
	Fold4	**80.89**	74.1	68.5	75.47
	Fold5	**85.56**	81.72	61.5	78.72
	Average	**83.14**	78.44	67.13	80.98

* The best accuracy in each row is highlighted using bold font.



(a) Seed = 42



(b) Seed = 77

Figure 12. Results of K-fold cross-validation of four efficient CNNs using two random seeds to split data in each fold.

As shown in Table 7 and Figure 12, MobileNetV1 (with input shape 224 × 224 × 3) produces the highest average accuracy. Although EfficientNetB0 could also achieve competitive performance, its model size and parameters are much larger than MobileNetV1. In future experiments, MobileNetV1 will be selected for the backbone network for tumor classification.

4.2.3 Weighted Binary Cross-Entropy

In CAD systems for disease diagnose, sensitivity is more important than other metrics. Sensitivity measures a model's ability to find out disease cases from normal cases. Poor sensitivity of a model will result in the missing classification of many positive cases to negative cases (high false negative). In breast cancer diagnosis, it is intolerant to misclassify a malignant tumor to a benign tumor because the false positive will lead to the late detection of cancer and extremely lower the survival chance of patients.

To make a tradeoff between the sensitivity and specificity while not to decrease the overall accuracy of the model, we use the weighted binary cross-entropy (WBCE) as the loss function. The WBCE is defined by

$$\boldsymbol{L}_{WBCE} = -\frac{1}{\mathrm{M}} \sum_{i=1}^{M} \left(\boldsymbol{W}_p \cdot \boldsymbol{Y}_i \cdot \log(\boldsymbol{H}_i) + (\mathbf{1} - \boldsymbol{Y}_i) \cdot \log(1 - \boldsymbol{H}_i) \right) \qquad (9)$$

where M denotes the number of training examples; $\boldsymbol{Y}_i$ is the target label of the *i*th training sample; $\boldsymbol{H}_i$ is the prediction; and the weight $\boldsymbol{W}_p$ is added as a multiplicative coefficient for the positive labels term. The additional weight $\boldsymbol{W}_p$ with value large than 1 will increase the loss for false negative results, hence minimizing the loss function will increase the sensitivity.

The effectiveness of the additional weight added in the loss function is tested using nine different weight values from 1 to 5 with a step size 0.5. Except for the positive weight value in the WBCE loss function being different, all the other configurations for each training are controlled to be the same, including the data preparation, model weight initialization, learning rate, momentum in SGD optimizer, total training epoch (100), and the batch size (32). All these experiments are performed using CPU.

Figure 13 plots values of the accuracy (ACC), sensitivity (SEN) and specificity (SPE) calculated using the test set for different weights. Compared with the standard BCE loss function where the positive weight is 1, the sensitivity gets improved if the weight increases. A higher weight value for the positive label term helps strongly penalize the False Negatives and compensate the class imbalance problem.

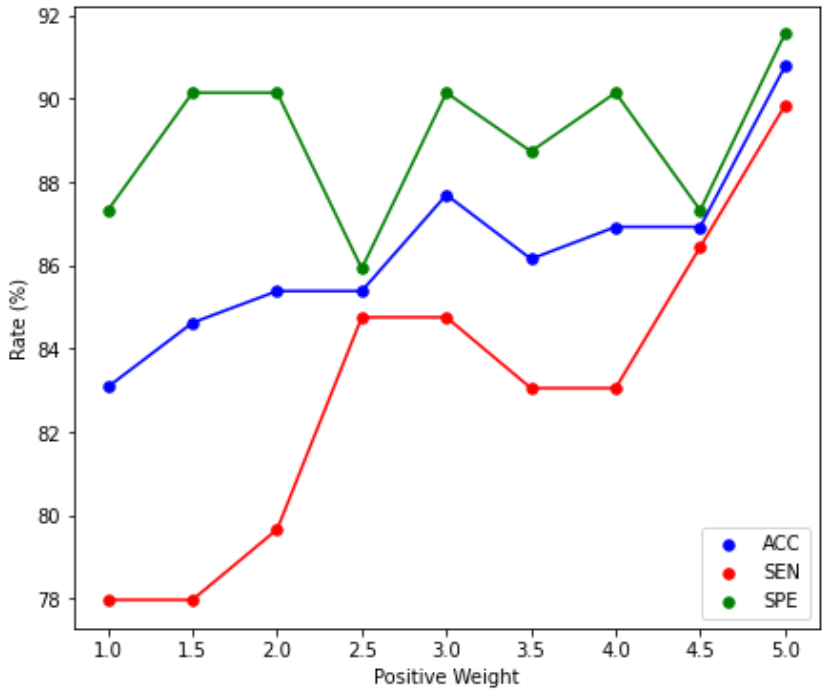


Figure 13. ACC, SEN and SPE of tumor classification using different weights in the WBCE loss function.

4.2.4 Classification Using Image Augmentation

The total number of BUS images is limited. After splitting 30% of images to be the validation set and testing set, the training dataset is relatively small, with only 604 images. Training a CNN using a small training set will highly likely lead to overfitting and not generalizing well from the validation data to test data. We explore the effectiveness of

several image augmentation techniques to improve the generalization of the tumor classification.

In breast cancer detection, tumor shapes and boundaries are important to determine the tumor categories. Augmentation techniques are not allowed to cause any tumor distortion, permutation of the relative position of image pixels, and new white pixels (confuse with calcification). Therefore, techniques including random rotation, rescaling, shearing, zooming, vertical flip, and brightness shifting are applied in our experiments. The final augmentation experiments are evaluated on traditional transformation techniques, including width shifting, height shifting, and horizontal flip. Figure 14 shows the example of an original resized image and its corresponding augmented images using width shifting by 10% of the image size, height shifting by 10%, and horizontal flip.

K-fold (K=5) cross-validation is performed on the CPU to measure the effectiveness of the three image augmentation techniques. Two seeds, 42 and 77, are used for splitting the train, validation, and test sets twice. The results are shown in Table 8 and Figure 15. Among the three techniques, width shifting achieved better overall accuracy than the other two techniques. Using the height shifting faces with risk of decreasing the accuracy.

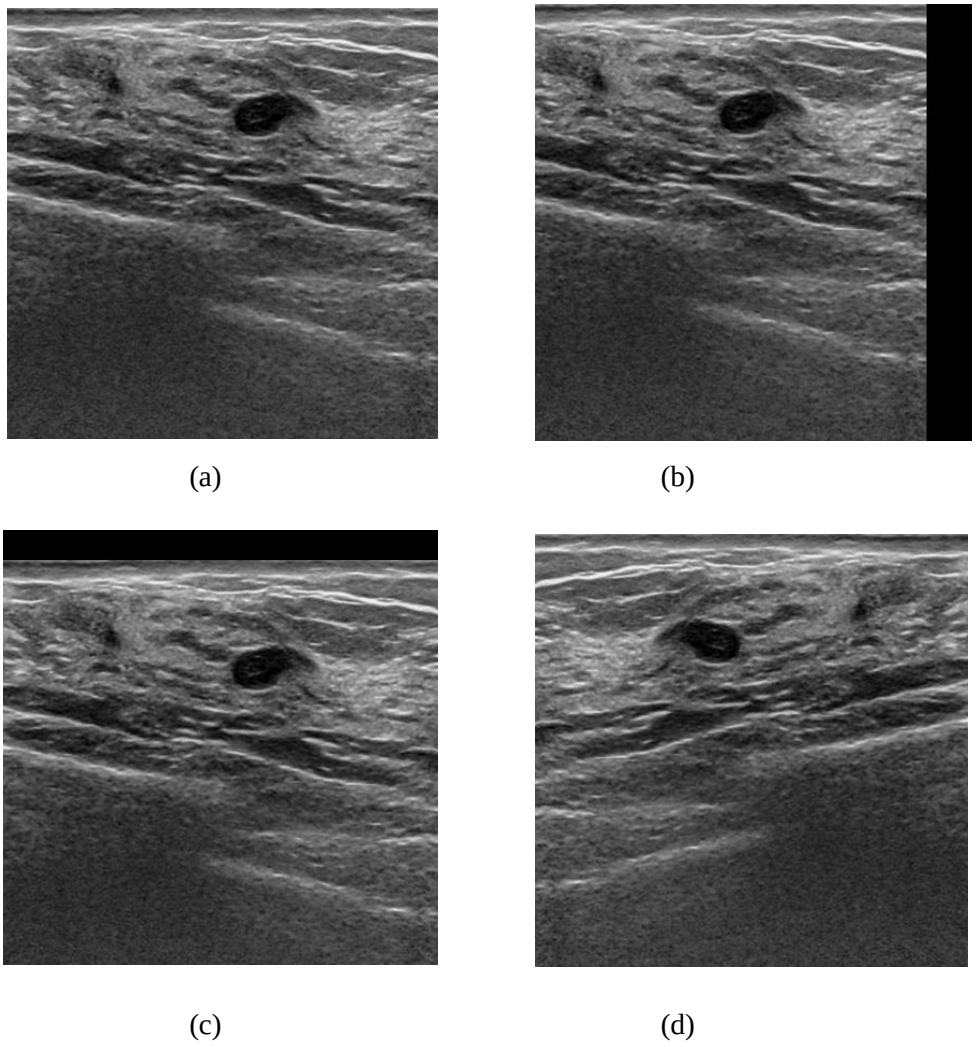
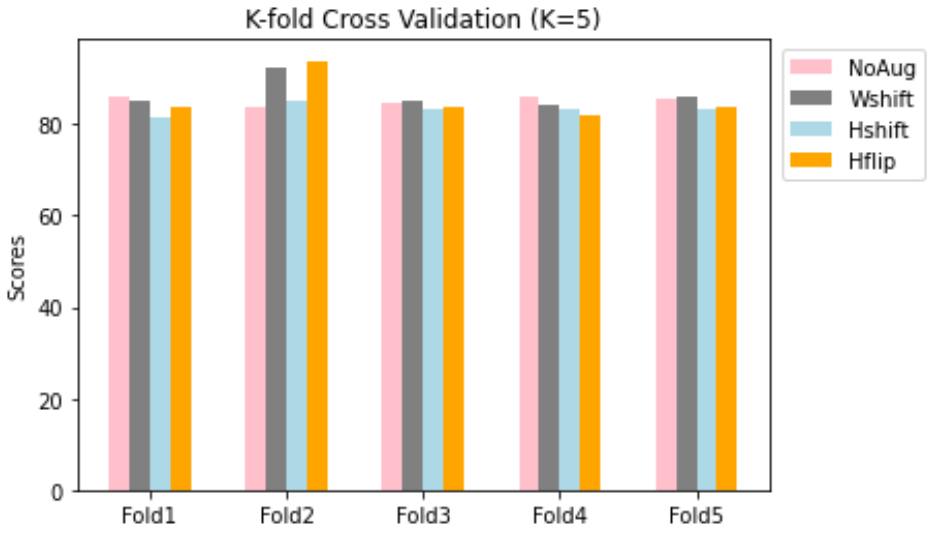


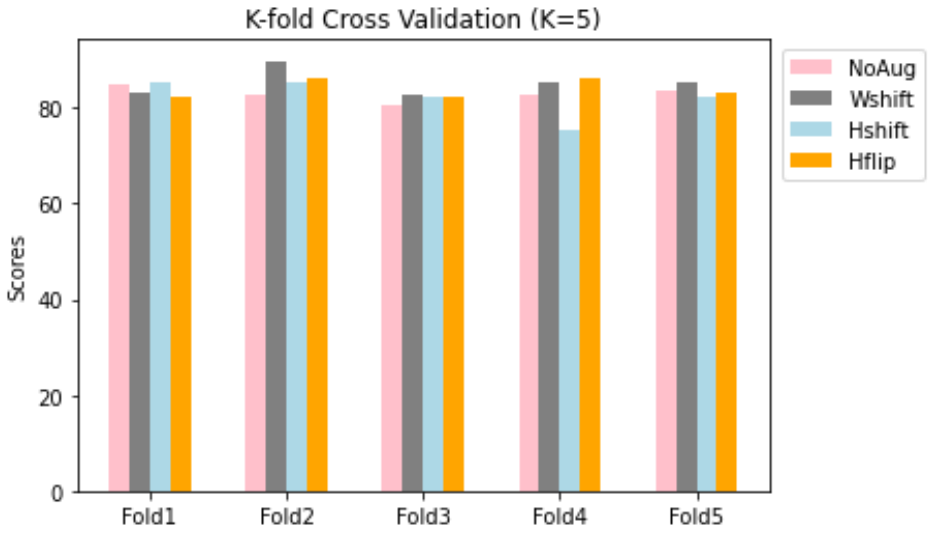
Figure 14. Example of (a) an original resized image with (b) width shifting by 10%, (c) height shifting by 10%, and (d) horizontal flipping.

Table 8. Tumor classification using various augmentation techniques.

		No Aug	**Width Shift**	**Height Shift**	**Horizontal Flip**
Seed = 42	Fold1	**86.15**	85.19	81.48	83.67
	Fold2	83.69	92.59	85.19	**93.88**
	Fold3	84.62	**85.19**	83.33	83.67
	Fold4	**86.15**	84.26	83.33	82
	Fold5	85.38	**86.11**	83.33	83.67
	Average	85.20	**86.67**	83.33	85.38
Seed = 77	Fold1	85.06	83.33	85.19	**87.23**
	Fold2	82.76	**89.81**	85.16	82.41
	Fold3	80.46	82.81	82.33	**86.04**
	Fold4	82.76	85.2	75.53	82.41
	Fold5	83.72	85.18	82.11	**86.11**
	Average	82.95	**85.27**	82.06	84.06



(a) Seed = 42



(b) Seed = 77

Figure 15. K-fold ACCs of the tumor classification using different augmentation techniques.

4.3 Multi-task Tumor Classification and Segmentation

4.3.1 Single-task Efficient Tumor Segmentation

U-Net is one of the most commonly used CNN for medical image segmentation using the concept of deconvolution and built upon the elegant architecture of FCN. Although using U-Net could achieve satisfying performance, it does not focus on utilizing the parameters

efficiently and is not suitable for deployment on mobile devices. A new segmentation architecture is proposed to use Link-Net as the decoder with pretrained light-weighted CNN (MobileNetV1, MobileNetV2, and EfficientNetB0) as the encoder. Table 9 shows the size and number of parameters for each model.

Controlled experiments are conducted to measure the performance of those four introduced segmentation models on the cropped images. All configurations are the same. From results showing in Table 10 and Figure 16, compared with U-Net, Link-Net with the pretrained EfficientNetB0 achieves the best overall performance, and Link-Net with the pretrained MobileNetV1 achieves the second-best overall performance. However, the pretrained EfficientNetB0's IOU is only 2.3% higher than that of the MobileNetV1, and it consumes 51.8% more memory storage. Our final model will be deployed on the mobile device, and the slight decrease of performance could be tolerated if the efficiency of segmentation could be largely improved. Based on this criterion, the Link-Net with the pretrained MobileNetV1 will be used in our final model.

Table 9. Size and Number of Parameters of evaluated segmentation models

Model	**Size**	**Parameters**
U-Net	89 MB	7,760,097
Link-Net with Pretrained MobileNetV1	54 MB	4,546,065
Link-Net with Pretrained MobileNetV2	50 MB	4,144,577
Link-Net with Pretrained EfficientNetB0	82 MB	6,096,333

Table 10. Performance of the 4 evaluated segmentation models

Model	**DSC**	**IOU**	**TPR**	**TNR**	**ACC**	**AUC**
U-Net	87	79.5	88.2	98.8	97.2	94.8
Link-Net with Pretrained MobileNetV1	86.6	79.2	87.4	**99.4**	**98.4**	**95.7**
Link-Net with Pretrained MobileNetV2	86.3	78.8	86.9	99.1	97.3	**95.5**
Link-Net with Pretrained EfficientNetB0	**88.2**	**81**	**89.1**	99	97.6	**95.7**

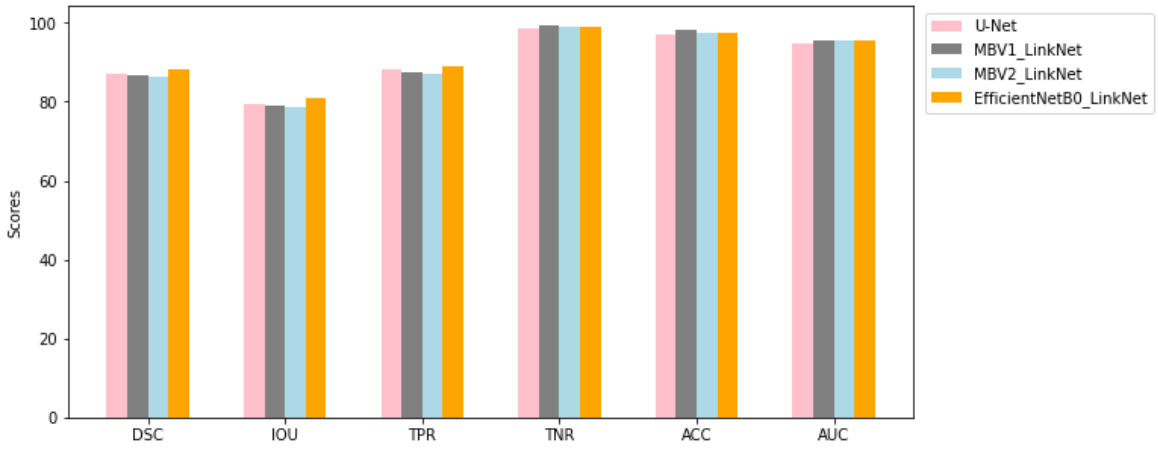


Figure 16. Performance Comparison Results of different segmentation models

4.3.2 Efficient Multitask Tumor Classification and Segmentation

Based on the single-task classification and segmentation, to construct a joint classification and segmentation model, the pretrained MobileNetV1 without its original dense layers is used as the network backbone in the multitask learning architecture. The feature maps from the backbone network are feed into (1) the newly introduced binary classification branch and (2) the Link-Net decoder segmentation branch.

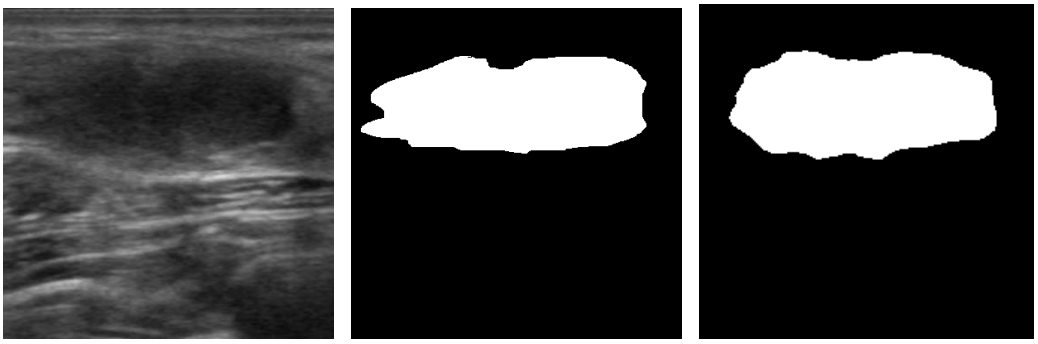
Firstly, controlled experiments are performed to evaluate the classification and segmentation performance of the constructed multitask model. The metrics DSC and IOU are used for the segmentation task; and ACC, SEN, and SPE are used for the classification task. The optimizer for the multitask model employs the Adam algorithm with a learning rate of 0.0005. The loss function for the classification branch is the Standard BCE, and the Dice loss is used for the segmentation branch. The ratio of the two losses in the final loss function is 1:1, which makes the two tasks equally important.

As shown in Table 11, the proposed multitask learning network effectively boosts both the classification and segmentation performance; and the size of the multitask model is only

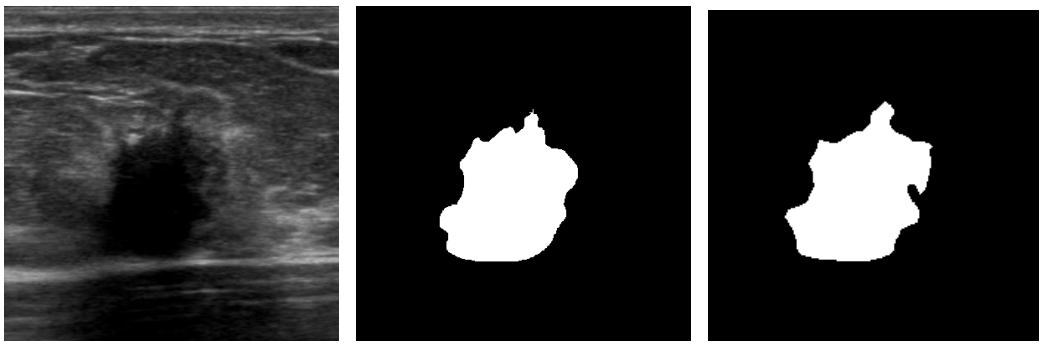
71.4% of the size of the summation of the single-task classification and segmentation model (84 MB). Figure 17 gives two examples for benign and malignant tumors with their mask and segmentation results from the multitask network.

Table 11. Segmentation and classification performance of proposed multitask network

Training	Size	Parameters	DSC	IOU	ACC	REC	SPE
Multitask network	60MB	5,136,658	**87.2**	**79.7**	**88.69**	**85.19**	**90.47**
Single-task Classification	30MB	3,819,457	-	-	83.85	79.66	87.32
Single-task Segmentation	54MB	4,546,065	86.6	79.2	-	-	-



(a) Benign case



(b) Malignant case

Figure 17. Examples of two original image (left), ground truth (middle), and segmentation results (right) from the multitask model.

Since classification is the major task in the multitask network. Controlled experiments are performed by assigning a larger weight of the BCE loss for classification. Table 12 shows the comparison for classification and segmentation, which shows that assigning large weight

for classification makes training pays more attention to optimize this task and could improve the classification performance.

Table 12. Segmentation and classification performance of different weight ratios.

Weight Ratio of Classification and Segmentation Loss	DSC	IOU	ACC	SEN	SPE
1:1	87.2	79.7	88.69	85.19	90.47
2:1	85.9	78.5	90.77	87.04	93.42
3:1	86.6	79	**93.85**	**94.44**	**93.42**

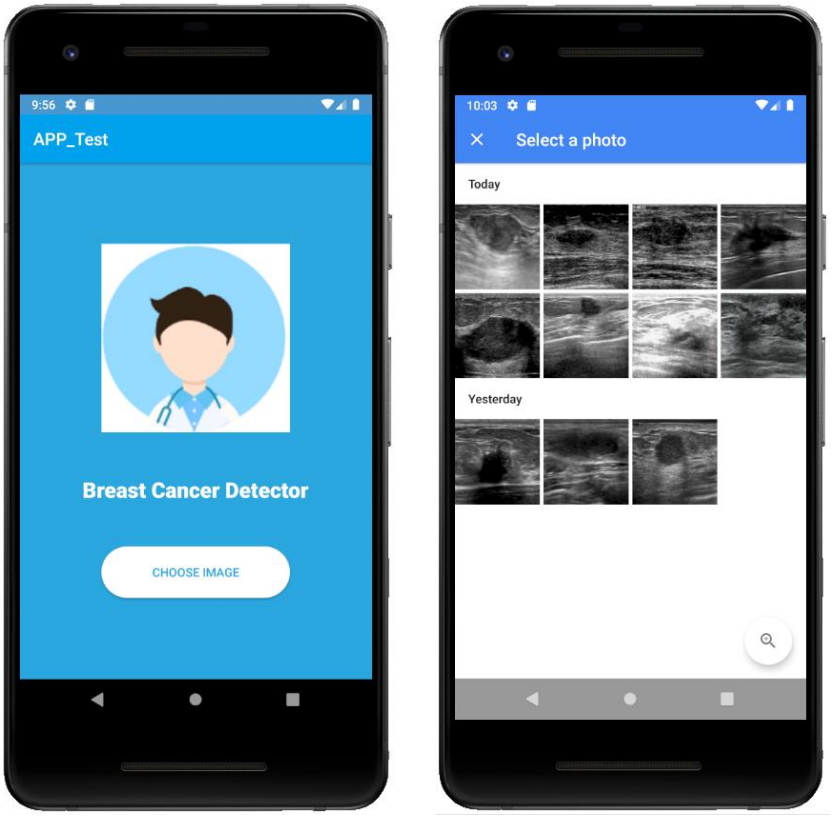
4.4 Mobile Application Development Using Android Studio

All the model's architecture and learnable weights are saved in the standard Keras H5 file format. We firstly convert it to TensorFlow Lite (. tflite) using the TFLiteConverter function provided by Tensorflow to use the TensorFlow Lite Java API for model optimal inference on Android.

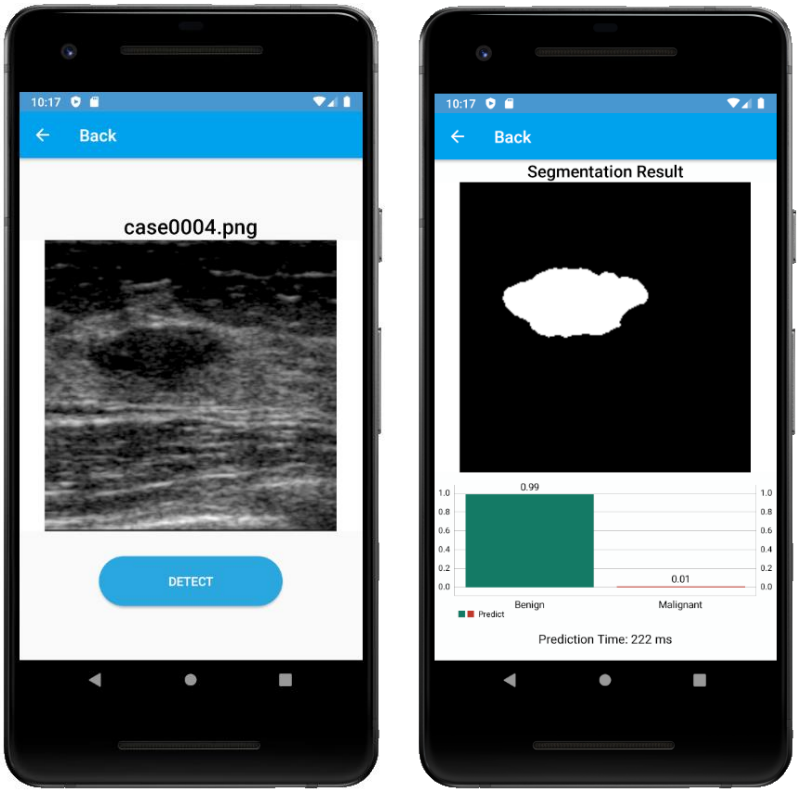
The mobile application is created and developed using Android Studio 3.2 installed on a Windows machine, with Android SDK with version 23 in the configurations. Since we do not have a physical Android device, we use Android Emulator to simulate an Android mobile device on the computer. It provides almost all the capabilities of a physical Android device. Google Pixel 2 API 28 with CPU x86 and 9.2 GB size on disk is downloaded as the instance of the Android virtual device (AVD), where our application is installed and tested.

Figure 18 shows the overall workflow of the developed mobile "Breast Cancer Detection" application. After launching the detection application, clicking the button "Choose Image" will allow the app access to the mobile photo gallery; after selecting a breast ultrasound images from the gallery, a page showing both the selected image will come up; clicking the "Detect" button at the bottom will enable the model starting the inference; the final result

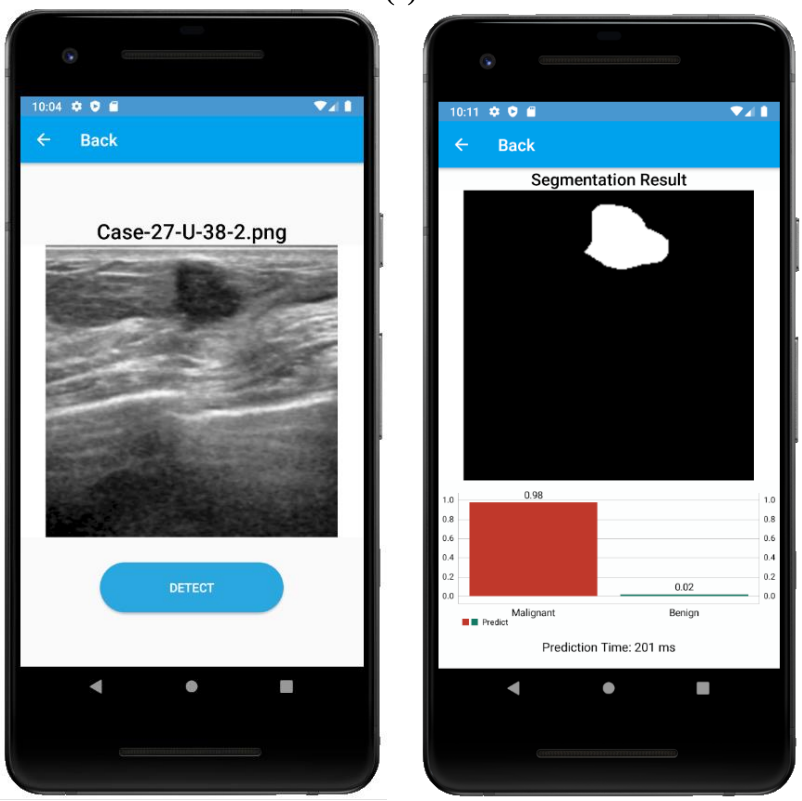
page covers three results: 1) segmentation result at the top; 2) classification results with a probability of malignancy; and 3) inference time cost (measured in microseconds).



(a) (b)



(c)



(d)

Figure 18. Mobile Application workflow. (a) landing page; (b) mobile photo gallery; (c) a benign case and the results; and (d) a malignant case and the results.

CHAPTER 5: CONCLUSION

The thesis focuses on developing and evaluating efficient and light-weighted CAD approaches for breast cancer detection, and building and deploying a mobile application using Android Studio. The major conclusions are as below.

(1) Padding and cropping preprocessing strategies are applied to prepare square input images of the tumor segmentation and classification networks. Experimental results demonstrate that the cropping strategy produces higher classification accuracy than the padding strategy.

(2) 4 pretrained light-weighted CNNs as classification models are measured from the accuracy perspective, including MobileNetV1, MobileNetV2, NASNetMobile, and EfficientNetB0. From the experiment results, MobileNetV1 achieves highest accuracy among the 4 evaluated classifiers.

(3) Image augmentation techniques including of width-shift, height-shift and horizontal-flip are employed to ease overfitting. Shown from the experiments, the width-shift with 0.1 percent of image size mostly improves the classification accuracy, compared with height-shift and horizontal-flip.

(4) From the experiments to evaluate the single-task segmentation models, including U-Net, Link-Net with pretrained MobileNetV1, MobileNetV2 and EfficientNetB0, Link-Net with pretrained MobileNetV1 outperforms with best tradeoff between segmentation accuracy and model efficiency.

(5) Testing results from multitask learning models prove that combining classification and segmentation task into a joint network improves the prediction accuracy of both tasks. The

classification task archives further better performance with larger weight assigned to the classification loss.

(6) The mobile App “Breast Cancer Detector” uses the multitask learning model converted in Tensorflow Lite format. The App supports real-time classification and segmentation of breast tumor simultaneously, with around 350 milliseconds average inference time.

References

[1] K. Czarnecki, J. M. Thompson, R. Seime, Y. E. Geda, J. R. Duffy, and J. E. Ahlskog, "Functional movement disorders: Successful treatment with a physical therapy rehabilitation protocol," *Parkinsonism & Related Disorders*, vol. 18, pp. 247–251, March 2012.

[2] U.S. Cancer Statistics Working Group, "United States Cancer Statistics: 1999–2008 Incidence and Mortality Web-based Report," *Atlanta (GA): Department of Health and Human Services, Centers for Disease Control*

[3] L. Fan, K. Strasser-Weippl, J.-J. Li, J. St Louis, D.M. Finkelstein, K.-D. Yu, W.-Q. Chen, Z.-M. Shao, and P.E. Goss, "Breast cancer in China," *The Lancet. Oncology*, vol. 15, e279-e289, 2014.

[4] H.D. Cheng, J. Shan, W. Ju, Y.H. Guo, and L. Zhang, "Automated breast cancer detection and classification using ultrasound images: a survey," *Pattern Recognition*, vol. 43, pp. 299-317, 2010.

[5] K. Doi, "Computer-aided diagnosis in medical imaging: historical review, current status and future potential," *Computerized medical imaging and graphics: the official journal of the Computerized Medical Imaging Society*, vol. 31, pp. 198–211, 2007.

[6] J. H. Kim, J. H. Cha, N. Kim, Y. Chang, M-S Ko, Y-W Choi, and H. H. Kim, "Computer-aided detection system for masses in automated whole breast ultrasonography: development and evaluation of the effectiveness," *Ultrasonography*, Vol. 33, pp. 105–115, April 2014.

[7] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, pp. 84-90, May 2017.

[8] T. Fujioka, K. Kubota, M. Mori et al. Y. Kikuchi, L. Katsuta, M. Kasahara, G. Oda, T. Ishiba, T. Nakagawa, and U. Tateishi, “Distinction between benign and malignant breast masses at breast ultrasound using deep learning method with convolutional neural network,” *Japanese journal of radiology*, vol. 37, pp. 466–472, June 2019.

[9] K. W. Wan, C. H. Wong, H. F. LP, D. Fan, P. L. Yuen, H. Y. Fong, M. Ying, “Evaluation of the performance of traditional machine learning algorithms, convolutional neural network and AutoML Vision in ultrasound breast lesions classification: a comparative study,” *Quantitative imaging in medicine and surgery*, vol. 11, pp. 1381-1393, 2021.

[10] J. Shi, S. Zhou, X. Liu, Q. Zhang, M. Lu, and T. Wang, “Stacked deep polynomial network based representation learning for tumor classification with small ultrasound image dataset,” *Neurocomputing*, vol. 194, pp. 87–94, June 2016.

[11] E. Stoffel, A. S. Becker, M. C. Wurnig, M. Marcon, S. Ghafoor, N. Berger, and A. Boss, “Distinction between phyllodes tumor and fibroadenoma in breast ultrasound using deep learning image analysis.,” *European journal of radiology open*, vol. 24, pp. 165–170, 2018.

[12] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, and A. Rabinovich, “Going deeper with convolutions,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1-9, 2015.

[13] S. Han, H. K. Kang, J. Y. Jeong, M. H. Park, W. Kim, W. C. Bang, and Y. K. Seong, “A deep learning framework for supporting the classification of breast lesions in ultrasound images,” *Physics in medicine and biology*, vol. 62, pp. 7714–7728, 2017.

[14] H. C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, “Deep convolutional neural networks for computer-aided detection:

CNN architectures, dataset characteristics and transfer learning," *IEEE transactions on medical imaging*, vol. 35, pp. 1285–1298, May 2016.

[15] P. M. Cheng, and H. S. Malhi, "Transfer learning with convolutional neural networks for classification of abdominal ultrasound images," *Journal of digital imaging*, vol. 30, pp. 234-243, April 2017.

[16] B. Shareef, M. Xian, and A. Vakanski, "STAN: SMALL TUMOR-AWARE NETWORK FOR BREAST ULTRASOUND IMAGE SEGMENTATION," Proceedings. *IEEE International Symposium on Biomedical Imaging*, vol. 2020, pp. 1469-1473, April 2020.

[17] Q. Huang, Y. Luo, and Q. Zhang, "Breast ultrasound image segmentation: a survey,". *International journal of computer assisted radiology and surgery*, vol. 12, pp. 493-507, March 2017.

[18] M. Xian, Y. Zhang, H.D. Cheng, F. Xu, B. Zhang, and J. Ding, "Automatic breast ultrasound image segmentation: a survey," *Pattern Recognition*, vol. 79, pp. 340-355, July 2018.

[19] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, pp. 640-651, April 2017.

[20] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481-2495, December 2017.

[21] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, vol. 9351, pp. 234–241, 2015.

[22] Z. Zhuang, N. Li, A. N. J. Raj, V. G. V. Mahesh, and S. Qiu, “An RDAU-NET model for lesion segmentation in breast ultrasound images,” *PLoS one*, vol. 14, August 2019.

[23] Y. Hu, Y. Guo, Y. Wang, J. Yu, J. Li, S. Zhou, and C. Chang, “Automatic tumor segmentation in breast ultrasound images using a dilated fully convolutional network combined with an active contour model,” *Medical Physics*, vol. 46, pp. 215–228, January 2019.

[24] V. Kumar, J. M. Webb, A. Gregory, M. Denis, D. D. Meixner, M. Bayat, D. H. Whaley, M. Fatemi, and A. Alizad, “Automated and real-time segmentation of suspicious breast masses using convolutional neural network,” *PLoS one*, vol. 13, 2018.

[25] M. Xian, Y. T. Zhang, and H. D. Cheng, “Fully automatic segmentation of breast ultrasound images based on breast characteristics in space and frequency domains,” *Pattern Recognition*, vol. 48, pp. 485–497, 2015.

[26] F. Pasa, V. Golkov, F. Pfeiffer, D. Cremers, and D. Pfeiffer, “Efficient Deep Network Architectures for Fast Chest X-Ray Tuberculosis Screening and Visualization,” *Scientific Reports*, vol. 9, no. 1, pp. 6268, April 2019.

[27] R. Sebastian, “An overview of multitask learning in deep neural networks,” *arXiv*:1706.05098, 2017.

[28] K. Simonyan, and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv*:1409.1556, 2014.

[29] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *arXiv*:1512.03385 [cs.CV], 2016.

[30] G. Xie, J. Wang, T. Zhang, J. Lai, R. Hong, and G. J. Qi, “Igcv 2: Interleaved structured sparse convolutional neural networks,” *arXiv*:1804.06202 [cs.CV], 2018.

[31] J. Hu, L. Shen, and G. Sun, “Squeeze-and-excitation networks,” *arXiv*:1709.01507 [cs.CV], 2017.

[32] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, “Model compression and acceleration for deep neural networks: The principles, progress, and challenges,” *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, January 2018.

[33] J. Jin, A. Dundar, and E. Culurciello, “Flattened convolutional neural networks for feedforward acceleration,” *arXiv*:1412.5474 [cs.NE], 2014.

[34] M. Wang, B. Liu, and H. Foroosh, “Factorized convolutional neural networks,” *IEEE International Conference on Computer Vision Workshops (ICCVW)*, vol.1, pp. 545-553, 2017.

[35] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J.Dally, and K. Keutzer, “Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 1mb model size,” *arXiv*:1602.07360 [cs.CV], 2016.

[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *arXiv*:1704.04861 [cs.CV], 2017.

[37] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L-C Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks. mobile networks for classification, detection and segmentation,” *arXiv*:1801.04381 [cs.CV], 2018.

[38] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition”, *arXiv*:1707.07012 [cs.CV], 2017.

[39] M. Tan, and Q. V. Le, “Efficientnet: Rethinking model scaling for convolutional neural networks,” *arXiv*:1905.11946 [cs.LG], 2019.

[40] F. Chollet, “Xception: Deep Learning with Depthwise Separable Convolutions,” *Computer Vision and Pattern Recognition (CVPR)*, pp. 1800-1807, 2017.

[41] S. Ioffe, and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” *International Conference on Machine Learning*, vol. 37, pp. 448-456, 2015.

[42] A. Chaurasia, and E. Culurciello, “LinkNet: Exploiting encoder representations for efficient semantic segmentation,” *IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, 2017.

[43] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *arXiv*:1512.03385 [cs.CV], 2015.

[44] A. Amyar, R. Modzelewski, H. Li, S. Ruan, “Multitask deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation,” Computers in Biology and Medicine, vol. 126, November 2020.

[45] R. Smithuis, L. Wijers and I. Dennert, “Ultrasound of the Breast,” *Radiology Assistant*, Alrijne hospital in Leiderdorp, the Netherlands.

[46] S. G. Orel, N. Kay, C. Reynolds, and D. C. Sullivan, “BI-RADS categorization as a predictor of malignancy,” *Radiology*, vol. 211, pp. 845-850, June 1999.

[47] M. Xian, Y. Zhang, H. D. Cheng, F. Xu, K. Huang, B. Zhang, J. Ding, C. Ning, and Y. Wang, “A Benchmark for Breast Ultrasound Image Segmentation (BUSIS),” *arXiv*:1801.03182 [cs.CV], 2018.

[48] M. H. Yap, G. Pons, J. Martí, S. Ganau, M. Sentís, R. Zwiggelaar, A. K. Davison, and R. Martí, “Automated Breast Ultrasound Lesions Detection Using Convolutional Neural Networks,” *IEEE journal of biomedical and health informatics*, vol. 22, no. 4, pp. 1218–1226, 2018.

[49] A. Rodtook, K. Kirimasthong, W. Lohitvisate, and S. Makhanov, “Automatic initialization of active contours and level set method in ultrasound images of breast abnormalities,” *Pattern Recognition*, vol. 79, pp. 172-182, 2018.

[50] M. Lin, Q. Chen, and S. Yan, “Network In Network,” *arXiv*:1312.4400 [cs.NE], 2013.

[51] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting,” *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014.

[52] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural Networks*, vol. 12, pp. 145-151, January 1999.

[53] X. Glorot, and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” *Artificial Intelligence and Statistics*, vol. 9, pp. 249-256, May 2010.

[54] E. Eban, M. Schain, A. Mackey, A. Gordon, R. Rifkin, G.Elidan, “Scalable learning of non-decomposable objectives,” *Artificial Intelligence and Statistics*, vol.54, pp. 832-840, 2017.

[55] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional Architecture for Fast Feature Embedding,” *arXiv:1408.5093 [cs.CV]*, June 2014.