

**HAND BLOOD VESSELS PATTERN
RECOGNITION**

TAN XUAN QING

UNIVERSITI TUNKU ABDUL RAHMAN

HAND BLOOD VESSELS PATTERN RECOGNITION

TAN XUAN QING

**A project report submitted in partial fulfilment of the
requirements for the award of Bachelor of Mechatronics
Engineering with Honours**

**Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman**

May 2023

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature :  _____

Name : Tan Xuan Qing _____

ID No. : 1903950 _____

Date : 2 May 2023 _____

APPROVAL FOR SUBMISSION

I certify that this project report entitled “**Hand Blood Vessels Pattern Recognition**” was prepared by **TAN XUAN QING** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Mechatronics Engineering with Honours at Universiti Tunku Abdul Rahman.

Approved by,

Signature : 

Supervisor : Dr. Tee Yee Kai

Date : 19 May 2023

Signature : 

Co-Supervisor : Dr. Aloysius Yapp

Date : 19 May 2023

The copyright of this report belongs to the author under the terms of the copyright Act 1987 as qualified by Intellectual Property Policy of Universiti Tunku Abdul Rahman. Due acknowledgement shall always be made of the use of any material contained in, or derived from, this report.

© 2023, Tan Xuan Qing. All right reserved.

ACKNOWLEDGEMENTS

I would like to thank all individuals who have contributed to the successful completion of this project. First and foremost, I would like to express my gratitude to my research supervisor, Dr. Tee Yee Kai, for his invaluable advice, guidance, and patience throughout the development of this research. I am also deeply grateful to Dr. Aloysius Yapp and Dr. Hau Lee Cheun for their support of my URS project, which enabled me to complete this project as my FYP.

Furthermore, I would like to extend my appreciation to my loving parents and friends who provided me with encouragement and support throughout the course of my FYP study. My friends, including Jonathan, Yi Ler, and Alvin, were a constant source of support and encouragement, even during the busiest times of the trimesters. Special thanks to Yoong Simyee for providing professional guidance on the anatomical study.

Besides that, thank you Dr Tang from Nanjing University of Aeronautics and Astronautics for providing the NTUIFDB v1 datasets.

Once again, thank you to all those who have contributed to this project.

ABSTRACT

This study aimed to address the problem of difficult venous access by proposing a vein feature extraction algorithm for the forearm using transfer learning on a U-net model with EfficientNetB3 as the backbone. The limited availability of forearm Near Infrared (NIR) image datasets and the lack of vein feature extraction algorithms focused on the forearm part were the main research problems. To evaluate the proposed model, the NTUIFDB v1 dataset containing 250 NIR forearm images was used and the performance was measured using Dice Coefficient and Jaccard Index. The results showed that the proposed model achieved an 83.56% Dice Coefficient and 71.76% Jaccard Index, outperforming four handcrafted techniques tested and other pre-trained models. This research contributes to the field by being the first to implement transfer learning on the NTUIFDB v1 dataset and provides a baseline for future studies to improve the proposed model. The proposed algorithm could aid in improving the success rate of venipuncture in patients with difficult venous access, such as pediatrics, geriatrics, obesity, and dark skin tone patients.

TABLE OF CONTENTS

DECLARATION		i
APPROVAL FOR SUBMISSION		ii
ACKNOWLEDGEMENTS		iv
ABSTRACT		v
TABLE OF CONTENTS		vi
LIST OF TABLES		ix
LIST OF FIGURES		x
LIST OF SYMBOLS / ABBREVIATIONS		xiii
LIST OF APPENDICES		xiv
CHAPTER		
1	INTRODUCTION	1
1.1	General Introduction	1
1.2	Importance of the Study	1
1.3	Problem Statement	6
1.4	Aim and Objectives	7
1.5	Scope and Limitation of the Study	7
1.6	Contribution of the Study	8
1.7	Outline of the Report	9
2	LITERATURE REVIEW	10
2.1	Anatomical Structure of Upper Limb	10
2.2	Vein Visualization through Near Infrared Imaging	13
2.3	Forearm NIR Datasets	14
2.4	Veins Pattern Extraction	15
2.4.1	Repeated Line Tracking	16
2.4.2	Maximum Curvature	18
2.4.3	Principal Curvature	21
2.4.4	Wide Line Detector	24

	2.4.5 U-Net Convolutional Neural Network	27
2.5	Transfer Learning	30
2.6	Pre-trained Models	32
	2.6.1 ResNet	32
	2.6.2 ResNeXt	32
	2.6.3 VGG	33
	2.6.4 Inception	33
	2.6.5 DenseNet	34
	2.6.6 SEnet	35
	2.6.7 MobileNet	35
	2.6.8 EfficientNet	36
2.7	Summary	37
3	METHODOLOGY AND WORK PLAN	39
3.1	Introduction	39
3.2	Datasets Handling Process	40
3.3	Forearm ROI Extraction	42
	3.3.1 Automatic ROI Extraction	42
	3.3.2 Manual ROI Extraction	46
3.4	Image Preprocessing	47
3.5	Veins Feature Extraction Algorithms	48
	3.5.1 State-of-the-art Algorithms	48
	3.5.2 U-net Transfer Learning Model	50
3.6	Evaluation of Algorithm	51
3.7	Work Plan	52
3.8	Summary	54
4	RESULTS AND DISCUSSION	55
4.1	Introduction	55
4.2	Results	55
	4.2.1 State-of-the-art Algorithms	55
	4.2.2 U-net Transfer Learning Model	57
4.3	Discussion	63
	4.3.1 Performance of State-of-the-art Hand Vein Segmentation	63

4.3.2	Performance of U-net Transfer Learning Model	63
4.3.3	Performance Analysis of State-of-the-art Algorithms and the Proposed Models	69
4.4	Summary	70
5	CONCLUSIONS AND RECOMMENDATIONS	71
5.1	Conclusions	71
5.2	Recommendations for future work	71
	REFERENCES	73
	APPENDICES	84

LIST OF TABLES

Table 2.1: Classification of Superficial Veins Pattern.	12
Table 2.2: NIR Forearm Datasets Reviewed.	15
Table 2.3: Veins Pattern Segmentation Comparison of U-Net Model.	30
Table 2.4: Summary of Performance of Literature Method.	38
Table 3.1: Parameters of Feature Extraction Algorithm.	49
Table 4.1: Accuracy, Specificity, Recall, and Precision of the State-of-the-art Vein Segmentation Algorithm. The best achieved results are shaded in green and worst results are shaded in orange.	55
Table 4.2: Dice Coefficient and Jaccard Index of the State-of-the-art Veins Extraction Algorithm. The best achieved results are shaded in green and worst results are shaded in orange.	56
Table 4.3: Accuracy, Specificity, Recall, and Precision of the U-net Transfer Learning Model. The best achieved results are shaded in green and worst results are shaded in orange.	57
Table 4.4: Dice Coefficient and Jaccard Index of the U-net Transfer Learning Model. The best achieved results are shaded in green and worst results are shaded in orange.	59
Table 4.5: Samples of segmented veins generated by the top performer of each pre-trained model family.	60
Table 4.6: Training and Validation Performance of the Top Performer Pre-trained Model in Each Pre-trained Model Family.	65
Table 4.7: Performance Comparison between State-of-the-art Algorithm and the Proposed Methods, where only the best backbone model is selected.	69

LIST OF FIGURES

Figure 1.1:	Transillumination Device from Veinlite (Veinlite, 2022).	4
Figure 1.2:	Dorsal Hand Vein Captured Using NIR Light Source (Sahana D S et al., 2020)	5
Figure 2.1:	Anatomical Structure of Upper Limb (Drake, Vogl and Mitchell, 2018).	10
Figure 2.2:	Cubital Fossa (Drake, Vogl and Mitchell, 2018).	11
Figure 2.3:	Major Superficial Veins at Cubital Fossa (Kim, Park and Park, 2017).	11
Figure 2.4:	Superficial Veins Pattern (Lee et al., 2015).	12
Figure 2.5:	NIR Image Acquisition through Reflection (left) and Penetration (right) (Pan et al., 2019).	13
Figure 2.6:	NIR Images Captured Using Penetration (left) and Reflection (right) (Kauba and Uhl, 2018).	14
Figure 2.7:	Flowchart of Repeated Line Tracking Algorithm.	17
Figure 2.8:	Dark Line Detection (Miura, Nagasaka and Miyatake, 2004).	17
Figure 2.9:	Finger Veins Pattern Extracted Using Repeated Line Tracking (Miura, Nagasaka and Miyatake, 2004).	18
Figure 2.10:	Flowchart of Maximum Curvature Algorithm.	19
Figure 2.11:	Cross-Sectional View of Finger Veins (Miura, Nagasaka and Miyatake, 2007).	19
Figure 2.12:	Cross Section Profile, Curvature, and Score Against Z (Miura, Nagasaka and Miyatake, 2007).	20
Figure 2.13:	Finger Veins Pattern Extracted Using Maximum Curvature (Miura, Nagasaka and Miyatake, 2007).	21
Figure 2.14:	Flowchart of Principal Curvature Algorithm (Choi et al., 2009).	21
Figure 2.15:	Principal Curvature and Direction(Choi et al., 2009).	23
Figure 2.16:	Finger Veins Pattern Extracted Using Principal Curvature (Choi et al., 2009).	23

Figure 2.17:	Flowchart of Wide Line Detector Algorithm (Huang et al., 2010).	24
Figure 2.18:	Circular Mask on the Finger Veins Image (Huang et al., 2010).	25
Figure 2.19:	Finger Veins Pattern Extracted Using Wide Line Detector (Huang et al., 2010).	26
Figure 2.20:	U-Net Architecture of a CNN model (Ronneberger, Fischer and Brox, 2015).	27
Figure 2.21:	Morphological and Skeletonization Veins Mask (Marattukalam and Abdulla, 2020).	28
Figure 2.22:	Veins Segmentation Results from U-Net (Marattukalam and Abdulla, 2020).	29
Figure 2.23:	U-Net Segmented Veins (Left) and Ground Truth (Right) (Jing et al., 2021).	30
Figure 3.1:	Process Flow of the Study.	39
Figure 3.2:	Group I (on left) and Group II (on right) of NTUIFDB v1.	41
Figure 3.3:	Group III (on left) and Group IV (on right) of NTUIFDB v1.	41
Figure 3.4:	Group V of NTUIFDB v1.	41
Figure 3.5:	Distribution of NTUIFDB v1 Datasets.	42
Figure 3.6:	Flowchart of Forearm Edge Detection.	42
Figure 3.7:	Sobel Edge Detection on Direction: Vertical (left) and Horizontal (Right).	43
Figure 3.8:	Visualizing the Forearm Edge Detection Process.	44
Figure 3.9:	Illustration of Defining Floor Tile Edge.	44
Figure 3.10:	Visualizing the Process of ROI Extraction Using Binary Mask.	45
Figure 3.11:	Manual Forearm ROI Extraction on CVAT.	46
Figure 3.12:	Flowchart of Image Pre-processing.	47
Figure 3.13:	Architecture of DeCNN.	47

Figure 3.14: Visualizing Image Pre-processing.	48
Figure 3.15: Confusion Matrix.	52
Figure 3.16: Gantt Chart of FYP Part 1.	53
Figure 3.17: Gantt Chart of FYP Part 2.	54

LIST OF SYMBOLS / ABBREVIATIONS

CLAHE	contrast limited adaptive histogram equalization
CNN	convolutional neural network
CPU	central processing unit
EER	equal error rate
GPU	graphics processing unit
LED	light-emitting diode
NIR	near-infrared
ReLU	Rectified Linear Unit
ROI	region of interest

LIST OF APPENDICES

APPENDIX A: Images

84

CHAPTER 1

INTRODUCTION

1.1 General Introduction

The modern medical system has recorded that 80% of patients in the hospital have performed intravenous cannulation. Among the parts of the body, peripheral venous access is the most common type of access. However, most medical workers are still using the sense of touch and sight to perform venous access. Even experienced practitioners would make mistakes with patients who have indistinct blood vessels. Junior practitioners who lacked practice perform badly during their medical routine would cause side effects to the patients. Statistics showed that the US healthcare system spent an additional \$4.7 billion due to the difficult venous access problem (Balter et al., 2015). To solve this problem, near-infrared (NIR) imaging could visualize the blood vessel pattern for the medical practitioner during venous access. The good point of the NIR device is a portable, low-cost solution, and non-invasive method as an aid of venous access.

1.2 Importance of the Study

Intravenous cannulation and venipuncture are one of the most common invasive routines carried out during patient treatment and laboratory testing. In the conventional method, the physicians and nurses perform needle injections through palpation and observation which requires both skills and experience to complete. The practitioners first identify the vein's location and then insert the needle into the center of the blood vessels. However, there are various challenges in accessing the location and depth during the practical situation due to skin colour tone, thickness of subcutaneous fat, and ageing of the skin (Cheng et al., 2016; Ichimura, Sasaki and Ogino, 2020). Research showed that the failure rate of peripheral intravenous cannulation is recorded at 30% during the first needle injection (van Loon et al., 2018). This would cause the patients to be feared of the needles or more serious harm such as haemorrhage in the aftermath of needle injection failure (Walsh, 2008).

There is research found that only 22.6% of dentists in Malaysia were confident to perform intravenous cannulation on adults while the confidence rate dropped to 13% when the target was children (Mohd Hakim Bin Mohamed Ashri, 2021). Pediatric patients had small and thin blood vessels which unlike adult vessels are with texture and thick diameter. Children have fragile veins and are unable to stay still during the venous cannulation which might make them uncomfortable during needle injection. Failure of performing venous cannulation may cause severe cases and complications to the patients where there were 10% to 25% of cases had been reported (Gallieni, Pittiruti and Biffi, 2008). According to the data of the FDA, there were 52% of complications were caused by the skills of healthcare workers, 12% were due to equipment failure, and 30% of cases were unverifiable (Scott, 1995). Complications of failure venous access included 34% of infiltration and extravasation cases, 20% of phlebitis cases, and other cases such as infections, Catheter-related bloodstream infection (CRBSI), swelling of veins, etc (Armenteros-Yeguas et al., 2017). Difficult venous access has been a problem that caused trouble to the patients and medical practitioners too. Therefore, physicians would consider applying chemical or physical techniques to improve venous access probability during normal practice.

The first technique to introduce is using fist clenching which can increase the blood flow rate in the cephalic and ulnar veins as the blood volume in the vessels especially the dorsal would enlarge (Simons et al., 1996). It is proven that performing hand gripping exercises would dilate the vessels due to the muscarinic receptors under the human skin (Török et al., 1997). The drawbacks are significant as it requires a period to dilate the veins. Besides that, nitroglycerin (NTG) which is used to treat coronary artery disease is proven that can result in rapid dilation of the dorsal veins even on a patient's hand that can't be observed veins initially (Roberge et al., 1987). After applying the NTG ointment, there are two reference groups achieved 50% and 83% success rates of venipuncture during the first attempt. However, physicians who used NTG in venipuncture would suffer some side effects such as nausea, headache, and blurred vision if no gloves were worn (Anon., 2022). Moreover, Rhys-Davies Exsanguinator (RDE) and Esmarch Bandage (EB) are commonly used to fill the veins at the cubital fossa of patients in surgery to

exsanguinate the arm or leg. It was found that the cross-sectional area of the veins was enlarged by 0.32 cm^2 and 0.33 cm^2 using RDE and EB methods (Nee et al., 1994). However, there are risks to applying these methods to elder people or patients who used steroids that might cause skin tears. EB method would cause venous tortuosity and brittle when it's applied for more than 5 minutes (Mbamalu and Banerjee, n.d.).

There are alternative approaches to assist in performing venous access which is through imaging technology. It is a type of invasive technique that obtain blood vessel images to assist the medical practitioners in deciding the needle placement. The first introduced method is magnetic resonance imaging (MRI). Medical MRI utilizes the spinning behaviour of hydrogen nuclei in the water under a strong magnetic field to form a specific arrangement. Therefore, body tissues or substances consisting of water can be detected using MRI. The image of blood vessels can be taken rapidly with the high contrast of the blood vessels and their surrounding tissues. Through electrocardiogram (ECG) or pulse-trigger timing, the medical image could further differentiate the arteries and veins (Maruyama et al., 2007). Research performed MRI scans to carry out adrenal venous sampling (AVS) achieved 76.6% and 90.6% of success rates on the right and left adrenal veins (Chayovan, Limumpornpetch and Hongsakul, 2019). Although the vessel image obtained using MRI is quite prominent, the cost and scale of the MRI machine are not a suitable choice for treating a large number of patients. Patients who have metals in their bodies for instance pacemakers, artificial joints, screws implanted during surgery, etc would not be recommended to perform MRI (NHS, 2022).

Other than that, the transillumination technique is used to examine the venous location using fiberoptic lights during venipuncture. The light illuminates the patient's subcutaneous tissue and can pass through the fats to assist the practitioners to observe the superficial veins (Kuhns et al., 1975). The light source used would not cause thermal injury and is suitable to be applied to an infant's hand. VEINLITE device uses a U-shape design surrounded by several light emitting diodes (LEDs) to map the superficial veins when the device is close to the target side (Thomas P. Duffy, Gerald T. Harder and Kevin Mori, 2012). In a study using transillumination to perform

intravenous cannulation, there is an 80% of success rate in adult patients while 70% to 94% in pediatric patients (Atalay et al., 2005). The transillumination technique is much safer and the size of the device is much smaller than MRI and ultrasonic. However, the environment would require to be dimmed than the ambient light when using the device due to the wavelength of the light source is within the visible light region. To achieve a higher penetration rate, red or orange light LEDs are often used because it's the longest wavelength among the visible light. Light source from visible light range limits the veins' visibility of certain patients for instance darker colour tone patients where the light is more likely to be absorbed by melanin and an obese person where light is dispersed through the subcutaneous fat layer.



Figure 1.1: Transillumination Device from Veinlite (Veinlite, 2022).

Moreover, ultrasound is also part of the routine of medical practitioners to guide peripheral venous access. It uses a high-frequency range of sound waves which ranges from 2 MHz to 10 MHz. The pulse signal projected to the tissue layer is then reflected to the probe. The ultrasonic images will display the reflected signal based on its characteristics. Blood vessels will appear black in the ultrasonic images due to the full transmission of the ultrasound signal (Hatfield and Bodenham, 1999). An author compares the performance of ultrasound and transillumination when performing intravenous cannulation where they achieved 92.5% and 80% of success rates (Girgis, 2014). However, if only the first attempt is counted as a success, transillumination had only 57.5% of success rate while ultrasound has 82.5% of success rate. This shows that ultrasound is much more reliable for practitioners to make decisions.

Ultrasound shares the same drawbacks as MRI where the cost of equipment is high which is not suitable to perform on large numbers of patients.

Apart from the methods introduced above, near-infrared (NIR) imaging is widely used in the medical industry, especially in the characteristics of visualizing veins. It can penetrate deeper under the human skin than visible light to obtain higher contrast of image compared to the transillumination method. NIR wavelengths between 700 to 1000 nm are often used to capture the veins pattern due to the haemoglobin absorbing the electromagnetic spectrum within this range while other tissues do not (Cuper et al., 2011). Therefore, the blood vessels are different from other tissues in the NIR image, and the blood vessels look darker than the surrounding tissue.



Figure 1.2: Dorsal Hand Vein Captured Using NIR Light Source (Sahana D S et al., 2020)

Due to the NIR absorption characteristics of haemoglobin, it is a type of non-invasive technique that determine the haemoglobin level in the human body. NIR spectroscopy is one of the applications to monitor the brain function of patients. In fact, oxygenated haemoglobin and deoxygenated haemoglobin have a difference in absorption rates, clinicians can monitor the activity of the brain through the blood flow in the cerebral cortex (Mcintosh et al., 2010). The penetration of NIR is not easy to pass through the human head,

therefore the NIR signal is collected from its reflection of scattered signals. The benefit of NIR spectroscopy compared with other existing methods is that it doesn't generate noise during the patient's movement (Sakudo, 2016). It is then useful to observe the neural activity under the subject's movement.

In addition, NIR is being applied in biometrics verification by performing template matching with the real-time collected veins image and the database. The parts of the body that are commonly used to capture the vessel patterns such as the fingers, palm, dorsal, and iris (Kumar and Zhou, 2012). Veins pattern doesn't change over time which makes the template more reliable as compared to fingerprint would change due to chemical exposure. Blood vessel pattern is also difficult to be spoofed as the vessel are located underneath the skin which increases the security of the verification method. NIR blood vessel image collection process can be achieved without contact which makes it more feasible to be implemented after the pandemic of COVID-19 (Wu et al., 2019).

1.3 Problem Statement

Venous access is an essential aspect of medical procedures, particularly in cases that require repeated access or emergency interventions. Unfortunately, the current state of venous access is not optimal, with research indicating that up to 30% of patients experience failed needle injection during their first attempt, which can cause unnecessary discomfort, anxiety, and even trauma (van Loon et al., 2018). As cannulation is one of the most common invasive procedure done in the hospitals, this could lead to wastage of supplies such as needles, syringe, and alcohol swabs.

Additionally, difficult venous access issue is not only pose significant patient care challenges but also result in increased costs for the US healthcare system. Recent reports show that the difficult venous access problem in the US healthcare system has led to an additional \$4.7 billion in costs (Balter et al., 2015). The existing medical instruments such as MRI machines and ultrasound machines although provide high contrast and accuracy in vessel imaging, it is not easy to carry and inconvenient to use. These methods are not feasible to carry out during emergency situations, especially for the medical practitioners required to perform venous access outside of the hospital. In remote and rural

areas, the distribution of healthcare resources is far from the cities. Therefore, health workers cannot rely on advanced and expensive technology to perform needle injections. Low-cost but high-accuracy vein visualization tools are still lacking in the market.

In recent years, Near-Infrared (NIR) imaging technology has become increasingly prevalent, with applications across a range of industries, including medical imaging. While there has been extensive research into the use of NIR imaging for various medical applications, including vein feature extraction, there remains a significant gap in our understanding of forearm blood vessels. The forearm is a commonly used and relatively safe location for venous access procedures, such as blood drawing and intravenous fluid and drug infusions. In situations where peripheral venous access is unlikely to succeed, dorsal and central venous access are often used as substitutes (Waitt, Waitt and Pirmohamed, 2004). However, these procedures can be uncomfortable for patients, highlighting the need for improved methods for accessing forearm veins..

1.4 Aim and Objectives

This project is aimed to produce a forearm-based blood vessels pattern recognition. The vessel image should be visualized in the binary image where white pixels represent the blood vessel and black pixels are the background. Throughout the study, there are three objectives set to be achieved:

- i) To investigate the current state-of-the-art techniques of hand blood vessel feature extraction algorithm.
- ii) To develop a hand blood vessel feature extraction algorithm using transfer learning method.
- iii) To assess the performance of proposed approaches with respect to the state-of-the-art algorithms using the NTUIFDB v1 forearm dataset

1.5 Scope and Limitation of the Study

The tasks of this study can be briefly distributed into three parts. Firstly, all the forearm veins dataset used by the past researchers should be reviewed and the availability should be verified with the original author. Dataset with ground

truth will be prioritized as it will reduce the workload of the project. The ground truth data is used for performance evaluation purpose only.

Secondly, the region of interest of the forearms was segmented and further perform pre-processing enhancement of datasets. The blood vessels pattern in the raw datasets are less contrasted and contained unwanted backgrounds. This dataset must be enhanced before performing the next step.

Lastly, the feature extraction framework was chosen to identify the blood vessels in the datasets. The proposed algorithm was evaluated using the ground truth and compared with the literature proposed method.

Blood vessels datasets acquisition is removed from this study because it requires setting up a well-controlled environment to be less hassle for the image pre-processing stage. Data acquisition from the public would require registration with the university and therefore limited the progress arrangement of the study. Besides that, there are numerous applications of NIR blood vessel images as mentioned in Section 1.2, only blood vessel pattern extraction is being chosen as the purpose of the study. Therefore, the evaluation of the algorithm is based on the real location of the veins, unlike biometrics applications that required a template database for identity verification. In this study, the ground truth of the images is more critical to determining the performance of the proposed algorithm.

1.6 Contribution of the Study

This study makes notable contributions to the field of vein feature extraction. Specifically, it demonstrates the effectiveness of transfer learning for forearm vein segmentation using a limited dataset, with important applications in medical imaging. In addition, the study addresses a gap in the existing literature by focusing on vein feature extraction in the forearm, an area of research that has received limited attention despite its significance for applications such as vein finder and automated venipuncture devices. By providing valuable insights into this area, the study has the potential to enhance the accuracy and reliability of vein-based biometric systems.

1.7 Outline of the Report

Chapter 1 of this report presents a thorough examination of the importance of studying vein feature extraction algorithms. This chapter identifies problems in existing research and markets, and outlines the aims and objectives of this study.

Chapter 2 provides an extensive literature review on the topic of vein feature extraction, covering key areas such as the anatomical structure of the upper limb, vein visualization through Near Infrared Light, Forearm NIR dataset used by past researchers, vein pattern extraction algorithms, transfer learning on hand vein feature extraction, and open source pre-trained models.

Chapter 3 focuses on the methodology used in this study, including a detailed explanation of how the datasets were handled, how the forearm ROI was extracted, the pre-processing techniques applied to the datasets, the veins feature extraction algorithm, and the metrics used to evaluate the algorithm. Additionally, this chapter outlines the workplan for carrying out this research.

Chapter 4 presents the results obtained from the vein feature extraction model trained, and compares the state-of-the-art techniques with the proposed model to select the best algorithm.

Finally, chapter 5 ended up with a conclusion for this study and suggests possible directions for future research.

CHAPTER 2

LITERATURE REVIEW

2.1 Anatomical Structure of Upper Limb

The human's upper limb is made up of different systems such as bones, arteries and veins, muscles, and nerves. The forearm is the structural unit which connects the arm and hand through the elbow joint and wrist joint. The region between the anterior part of the elbow joint is called the cubital fossa. It is an imaginary triangular region in which the base is medial and lateral of epicondyles and the two sides of the triangle are brachialis muscles (Drake, Vogl and Mitchell, 2018). There are four major structures consisting in the cubital fossa which are (Bains and Lappin, 2021)

- i) Median nerve.
- ii) Lateral nerve.
- iii) Brachial artery.
- iv) Biceps tendon.

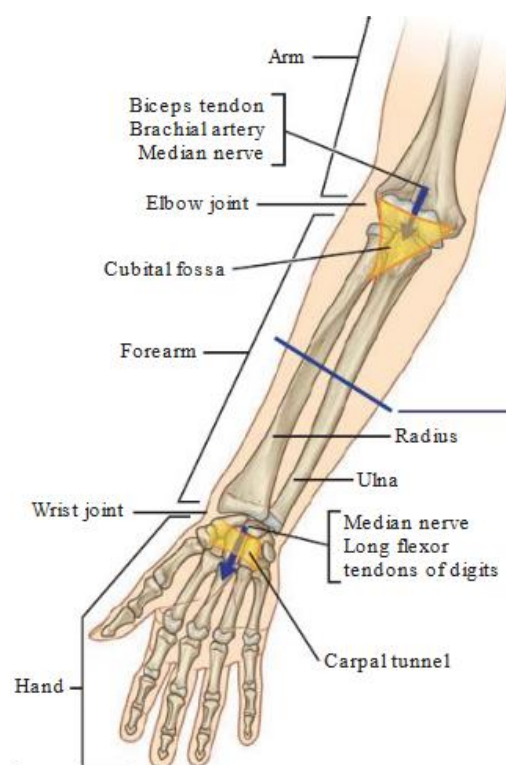


Figure 2.1: Anatomical Structure of Upper Limb (Drake, Vogl and Mitchell, 2018).

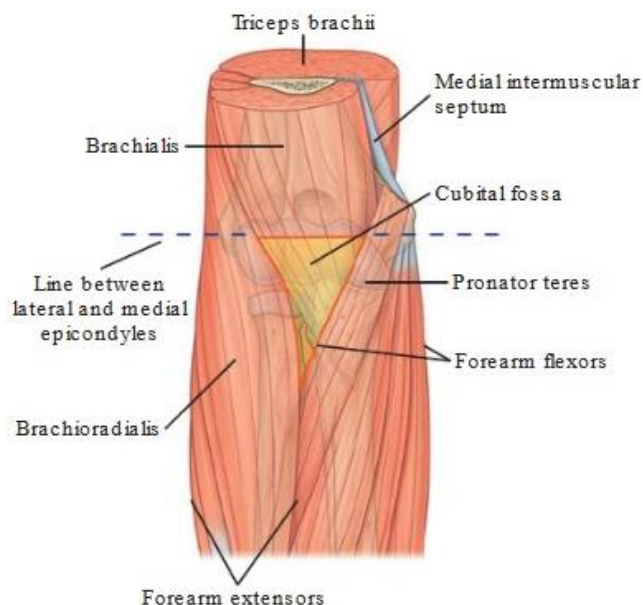


Figure 2.2: Cubital Fossa (Drake, Vogl and Mitchell, 2018).

The brachial artery is often used to measure the blood pressure level of the patients by placing the stethoscope over this region. Arteries are transferring oxygenated blood throughout various parts of the body which are at a higher pressure than veins. Therefore, clinical operations which involve needle injections would not consider arteries. Arteries in the cubital fossa are thinner and deeper than the veins and blood may spurt out due to the pressure in the vessels. On the other hand, veins have a larger diameter and thinner walls which carry 80% of the blood in the body (Moore and Rohen, 2013). There are numerous bicuspid valves which avoid the blood flow in the backward direction.

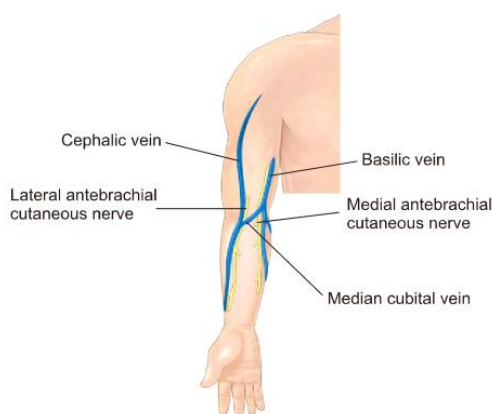


Figure 2.3: Major Superficial Veins at Cubital Fossa (Kim, Park and Park, 2017).

The veins at the cubital fossa can be classified into two types which are deep veins and superficial veins. The deep venous system on the upper limb stays closely with the arteries which are unable to be observed easily with naked eyes (Nguyen and Duong, 2021). Superficial veins are veins that can be accessed easily with palpation and observation. The major superficial veins at the cubital fossa are cephalic, basilic, and median cubital veins. The cubital fossa is an area that normally performed venous access because most of the superficial veins can be found in this region. The median cubital veins, among the superficial veins, serve as a connection point between the cephalic and basilic veins. This connection point is located at the cubital fossa and is considered a priority option for clinicians when performing venipuncture.

However, median cubital veins do not appear in the same form in all people's anatomical structures. According to the previous novel study, there are four different patterns as shown in Figure 2.4 (Lee et al., 2015).

Table 2.1: Classification of Superficial Veins Pattern.

Superficial Veins Type	Description
Type I	Median cubital vein is dominant after the cephalic and basilic veins merge at the node.
Type II	Median cubital vein branches across the cephalic and basilic veins.
Type III	Cephalic vein is missing or underdeveloped.
Type IV	Median cubital is missing while cephalic and basilic vein does not merge

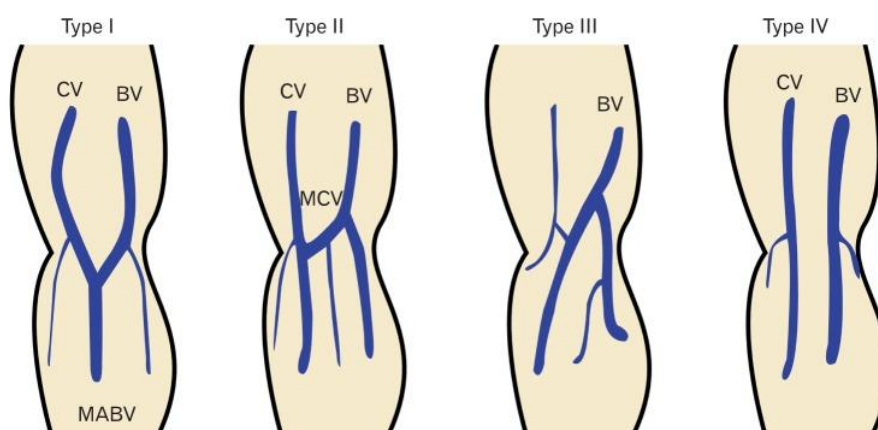


Figure 2.4: Superficial Veins Pattern (Lee et al., 2015).

Type I and II superficial veins pattern can be found in 96.8% of the patients which is the ideal case for performing venipuncture. It was found that type I and II are the pattern that normally described the median cubital veins in medical textbooks. Although type III and IV are rarely found in practical situations, they would cause confusion to medical workers who have less experience in performing venipuncture.

Superficial veins are often crossing with the nerves which are very close to each other (Horowitz, 2000). There were no patterns found that how nerves and veins intertwined each other, and it is very difficult to separate them. During the process of venipuncture, contact with the nerve and needle is possible to put the patients at risk of injury. Previous research found that there was no specific location of the anatomy that was the safest region for every individual (Yamada et al., 2008). However, it is suggested that the median cubital vein that is near the cephalic vein is unlikely to cause nerve damage. Cephalic and basilic could be an alternative if the median cubital vein is not suitable for venous access. The risk undertaken is that higher discomfort will be experienced due to nerve damage when venous access is performed at the basilic vein.

2.2 Vein Visualization through Near Infrared Imaging

There are two common techniques are being used to collect the NIR images of the veins pattern which are reflection and penetration, as shown in Figure 2.5. The camera is an infrared light-sensitive device that can receive the signal from the NIR region (Pan et al., 2019). Besides using the industrial camera, a USB camera equipped with an infrared transmission filter could capture NIR

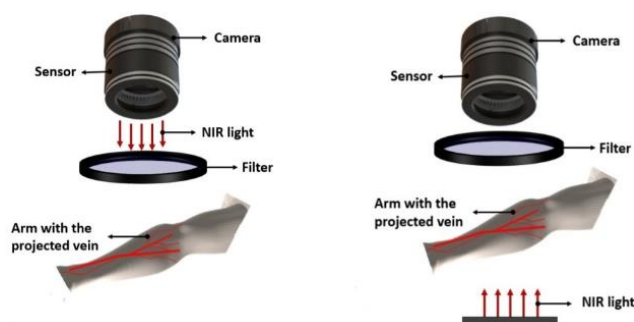


Figure 2.5: NIR Image Acquisition through Reflection (left) and Penetration (right) (Pan et al., 2019).

images as well (Crisan, Tarnovan and Crişan, 2007). For the reflection method, the NIR light source is located on the same side of the camera. When the emitted light touches the surface of the hand, a portion of the light is reflected the camera. However, the camera will receive environmental noises which lower the quality and contrast of the image. The advantage of this method is the device can be compact and small. Low power consumption can achieve due to the low intensity of light needed. On the other hand, the penetration method uses a larger device where the hand stays between the camera and NIR light source. The image captured comes with high contrast because most of the light that penetrates the hand is received by the camera. Therefore, datasets captured using the reflection method require better image processing techniques to obtain a clearer veins pattern result.

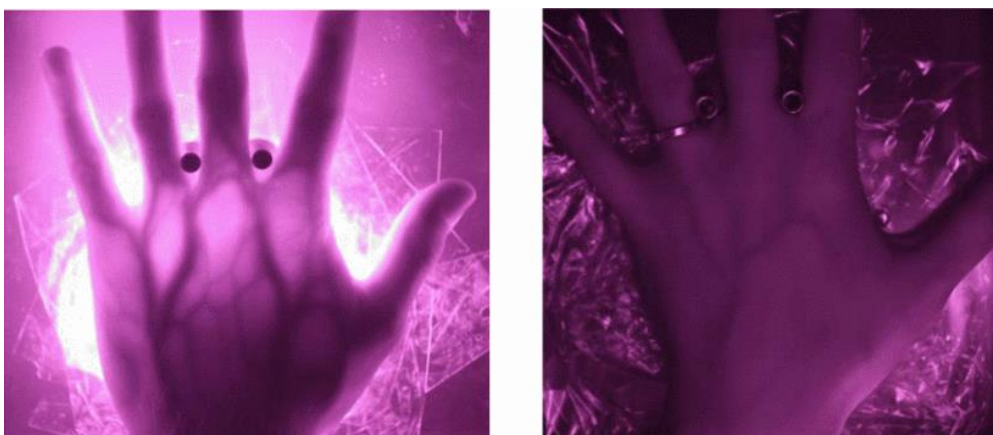


Figure 2.6: NIR Images Captured Using Penetration (left) and Reflection (right) (Kauba and Uhl, 2018).

2.3 Forearm NIR Datasets

Forearm NIR veins datasets were rarely mentioned in the research journal. Most of the NIR veins research were regarding biometrics application, where fingers, dorsal, and palmer were used due to the thin subcutaneous tissue. Forearm veins were not common body parts being used for biometrics purposes and therefore they could only be found in other research areas such as medical applications and forensic science. According to the summary of forearm datasets mentioned in the research papers as shown in Table 2.2, there were 3 datasets publicly available. The process of collecting the forearm veins is more challenging than the other parts like fingers because larger

environmental factors were required to control. The large forearm area often requires capturing the data under exposure to environment lighting which is also similar to the actual application environment of venous access. Ambient light would introduce noises to the image which would require more pre-processing techniques to obtain a clear image to proceed with veins feature extraction.

Table 2.2: NIR Forearm Datasets Reviewed.

Author	Year	No. of Datasets	Public or Private
NTUIFDB v1, Zhang et al., (2012)	2012	250	Public
NTUIFDB v2, Huynh et al. (2015)	2014	1312	Public
Azuetto-Ríos, Hernández-Gómez and Hernández-Santiago (2016)	2016	NaN	Private
Choras (2017)	2017	56	Private
VeinCV_RL Leli et al. (2020)	2020	320	Public (partially)
Shah et al., (2022)	2022	18	Private

2.4 Veins Pattern Extraction

There are two methods and approaches to extracting the veins pattern on the NIR image, which are computer vision techniques and deep learning techniques. The computer vision technique is a handcrafted feature extraction that can detect the veins through the characteristics of the image. As veins have different pixel values as compared to the background, this technique discovers this difference by following the instruction given. The computer vision technique usually has faster performance and is easier to implement. However, the drawback is it has lower compatibility to achieve similar performance on various datasets because the algorithm was normally designed based on the datasets collected by the author. The signal-to-noise ratio and background of the datasets will affect the performance of the algorithm. These

factors can be lowered by applying image pre-processing techniques to achieve a higher quality of images. Other than that, veins pattern extraction algorithms proposed by past researchers were mainly applied on fingers, dorsal, and palmer. The parameters of the algorithm may be suitable for fingers but not other parts like the forearm.

In recent years, deep learning techniques have become the mainstream to perform veins feature extraction. Unlike handcrafted feature extractions, a deep learning algorithm learns about the feature of the veins pattern through the training datasets. Tuning the parameters of the deep learning model can optimize the performance of the feature extraction. Deep learning techniques often achieve higher performance than computer vision techniques easily which is the reason researchers gained interest in the deep learning method. However, the deep learning model requires a lot of datasets so that the model can identify the feature of the veins. The NIR veins datasets that are publicly available are normally inadequate to train a deep learning model. Some researchers applied the transfer learning method which only requires a smaller number of datasets. Besides that, one significant drawback of the deep learning model is the high computational cost. GPU is often used instead of CPU which can parallel process a huge number of calculations.

2.4.1 Repeated Line Tracking

A repeated line tracking algorithm is designed to extract the veins pattern on the fingers for biometrics verification purposes (Miura, Nagasaka and Miyatake, 2004). It uses computer vision technology to identify the dark lines of the image which was aimed to achieve a feature extraction process in less than 1 second.

In the first step, the initial point is identified randomly, and the moving direction attribute is calculated to determine the direction of the next moving steps. Then, step 2 describes how the dark lines are identified using an illustration of the 's-p-t' valley. The current tracking point will go to point p which is its neighbour and identify the grey level of that point versus the two end points of the cross-sectional line, s and t. The relationship between the s, p, and t can be visualised on the graph shown in Figure 2.8. θ_i is the angle to adjust the neighbourhood reached which limits the line tracked from moving

away from the dark lines. Once the point at the deepest valley is identified, the point will become the current tracking position and the process is repeated until there is no valley formed. These tracking points identified will be stored in the locus space.

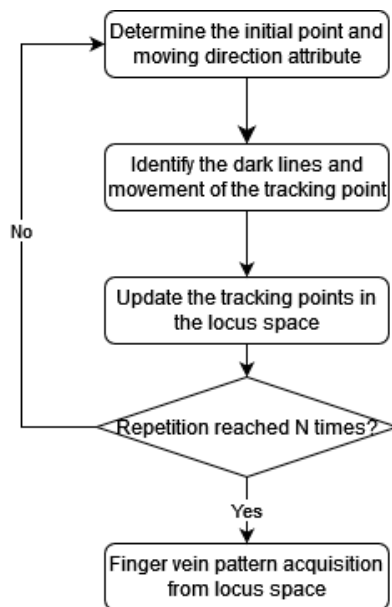


Figure 2.7: Flowchart of Repeated Line Tracking Algorithm.

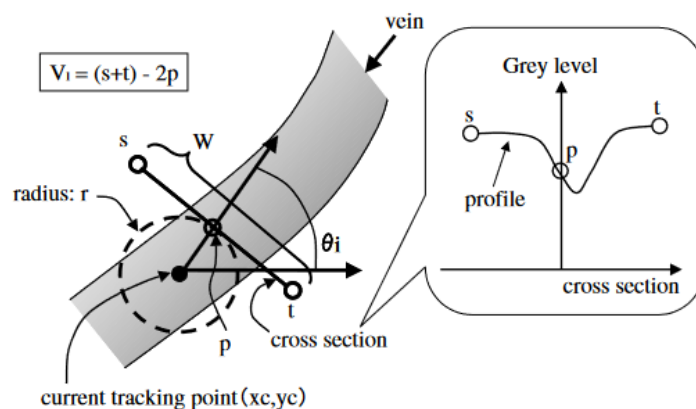


Figure 2.8: Dark Line Detection (Miura, Nagasaka and Miyatake, 2004).

Since the initial tracking point is identified randomly, the point outside the dark line may be identified. Noises in the image are also possible identified as tracking points. Therefore, the process is iterated N times. The points on the dark line will be detected more frequently than the noise which accumulates a higher value of it in the locus space. Although noises are possibly recorded as tracking points, the value will be comparatively lower

than real dark lines when the number of iterations becomes larger. Lastly, a thresholding method is applied to filter out the lower value of tracking points while a higher value of tracking points is identified as the veins.

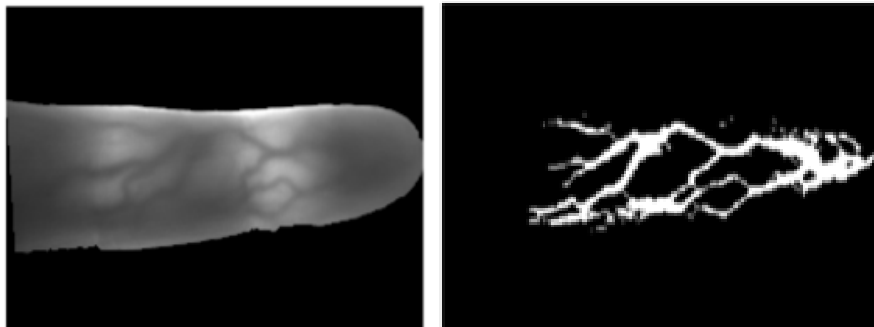


Figure 2.9: Finger Veins Pattern Extracted Using Repeated Line Tracking (Miura, Nagasaka and Miyatake, 2004).

The finger veins pattern extraction performance was evaluated using equal error rate (ERR). Repeated line tracking algorithm achieved 0.145% of EER. The feature extraction process took for average 450 ms using the author's hardware device. The datasets used for algorithm testing was collected using penetration method which the veins pattern has a good contrast. The image was only performed normalization before feature extraction to adjust the location and angle of the image captured.

2.4.2 Maximum Curvature

Maximum curvature algorithm is type of method that can solve the limitation of repeated line tracking, which is poor performance on extracting narrow veins (Miura, Nagasaka and Miyatake, 2007). Repeated line tracking relies on the accumulation values of tracking points and therefore wider lines would have larger tracking points while narrow lines has lesser. Maximum curvature finds the center position of the veins and connect them together. This method can still perform well under different brightness and vein width.

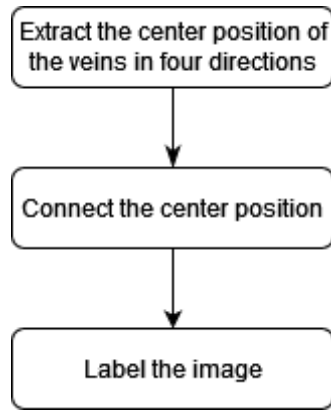


Figure 2.10: Flowchart of Maximum Curvature Algorithm.

In the first step of the algorithm, the center position of the veins are identified. The cross-sectional view, $P_f(z)$ of the finger veins as shown in Figure 2.11 describes the brightness level of the pixels, $F(x,y)$ in a graph. Point A, B, and C has lower brightness as the veins are darker than its surrounding pixels. Even the wide and narrow veins could be identified. The local minimum in the graph indicates that the dark veins are identified. The curvature of the cross-sectional profile is calculated as below.

$$k(z) = \frac{d^2 P_f(z)/dz^2}{\{1 + [dP_f(z)/dz]^2\}^{3/2}} \quad (2.1)$$

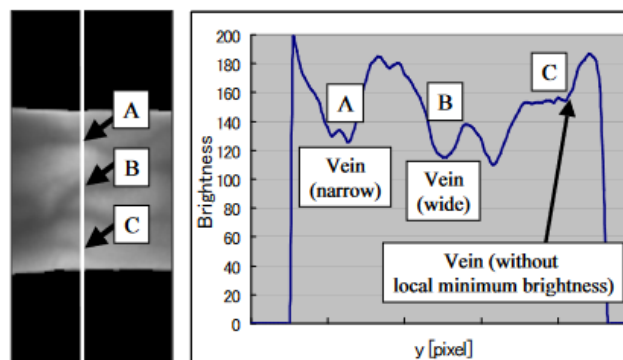


Figure 2.11: Cross-Sectional View of Finger Veins (Miura, Nagasaka and Miyatake, 2007).

The curvature, $k(z)$ function at the positive region is referred to as a concave area at the cross-sectional view. Obtaining the local maximum of $k(z)$

will return the center position of the veins. This local maximum is filtered by calculating the probability score, $S_{cr}(z)$ as below.

$$S_{cr}(z_i') = k(z_i') \times W_r(i) \quad (2.2)$$

The score includes the variable of curvature and width. Higher curvature is reflected by the brightness of pixels, where thin veins could possibly obtain a score. The larger width of the veins is also more likely to obtain a high score. The score designed has considered the thin and wide veins. The local maximum profile are obtained in four directions which are horizontal, vertical, and two 45° tilting lines of both horizontal and vertical directions. This can ensure that the feature extraction results don't change much although the vein's brightness and width are different in every image acquisition process. These four results were then combined by selecting the highest pixel values from four results.

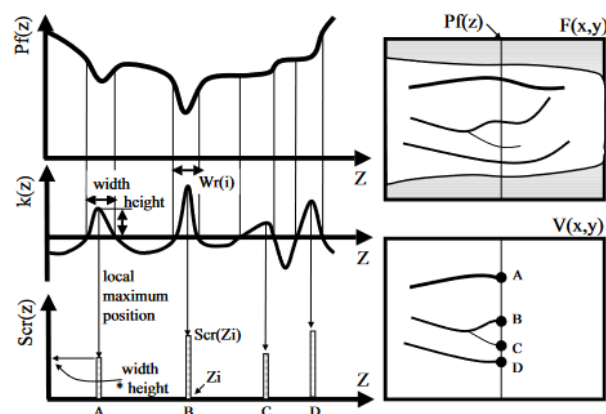


Figure 2.12: Cross Section Profile, Curvature, and Score Against Z (Miura, Nagasaka and Miyatake, 2007).

After the center position of the veins is identified, these centre points are connected to form a continuous line. The center points obtained contained different pixel values due to different brightness levels across the image forming gradient lines. Therefore, the higher pixel value will be lowered, and the lower pixel value should be tuned higher. Lastly, a thresholding method to convert the grey image become a binary image to remain the veins as white pixels.

The author performed veins feature extraction on the datasets and an EER of 0.0009% is achieved. The experiments also proved that maximum curvature achieved better performance during extracting the patterns of thin veins. The datasets were taken using the penetration method where the author does not perform any pre-processing techniques to enhance the datasets. Although the author claimed that maximum curvature performed well on lower brightness images, the statement is yet to be verified.

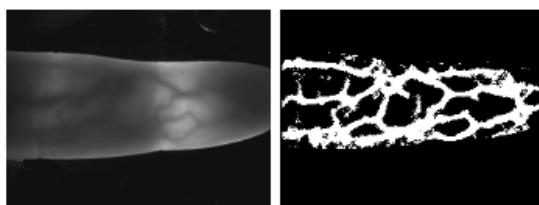


Figure 2.13: Finger Veins Pattern Extracted Using Maximum Curvature (Miura, Nagasaka and Miyatake, 2007).

2.4.3 Principal Curvature

Principal curvature is an advanced algorithm of the maximum curvature. It was found that the skin thickness and the veins size vary during every image acquisition process which would lead to false identified veins extraction (Sheng et al., 2011). Maximum curvature proposed that obtaining the four directions of local maximum would be unprecise enough when the image quality is not optimum. The author proposed gradient normalization to solve the uneven light illumination on the images which degrades the quality of the images. Principal curvature extracts the feature through the ‘true’ maximum curvature instead of the highest maximum curvature from only four directions (Choi et al., 2009).

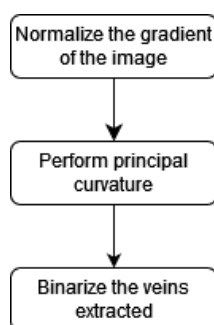


Figure 2.14: Flowchart of Principal Curvature Algorithm (Choi et al., 2009).

Image acquisition of hand veins often comes with noises from the environment. Due to the need of simplifying the image capturing device, uneven brightness is difficult to be controlled. The author proposed normalizing the gradient of the image to obtain an even gradient image. The gradient is expressed in vector form which stores the partial derivative of image intensity in the xy plane. A normalized gradient (which is also known as a unit vector) takes the gradient and divides it by its magnitude. Normalization gradient could achieve an even brightness and veins thickness image which can improve the image quality for feature extraction. After the normalization of the gradient vector, there are some noises with small gradient values in the image. These noises are assigned zero by applying a hard threshold value.

$$\mathbf{G}(x, y) = \nabla L(x, y) = \left(\frac{\partial L(x, y)}{\partial x}, \frac{\partial L(x, y)}{\partial y} \right) \quad (2.3)$$

$$\mathbf{G}_N(x, y) = \frac{\nabla L(x, y)}{\|\mathbf{G}(x, y)\|} \quad (2.4)$$

where

$L(x, y)$ = image intensity

$\mathbf{G}(x, y)$ = gradient

$\mathbf{G}_N(x, y)$ = normalized gradient

The veins feature can be extracted by applying the principal curvature. A Hessian Matrix is constructed which consists of second-order partial derivatives.

$$H(x, y) = \begin{pmatrix} \frac{\partial^2 L(x, y)}{\partial x^2} & \frac{\partial^2 L(x, y)}{\partial x \partial y} \\ \frac{\partial^2 L(x, y)}{\partial x \partial y} & \frac{\partial^2 L(x, y)}{\partial y^2} \end{pmatrix} \quad (2.5)$$

Then, the normalized gradient is applied with gaussian filter to remove the noises and smoothen the matrix. The eigenvalues and eigenvectors are then determined from the Hessian Matrix. The eigenvalues, λ_1 and λ_2 represents the

principal curvature; while eigenvectors, v_1 and v_2 indicates the direction of the local maxima and minima. In Figure 2.15, the maximum curvature, λ_1 and minimum curvature, λ_2 consist of v_1 and v_2 to indicate the direction. The local maxima obtained from Hessian Matrix is optimized and true according to the vein's images. The local maxima obtained from maximum curvature algorithm might not be optimized and inconsistent about the maxima calculated. Lastly, the veins feature extracted is binarized where the background is assigned as 0.

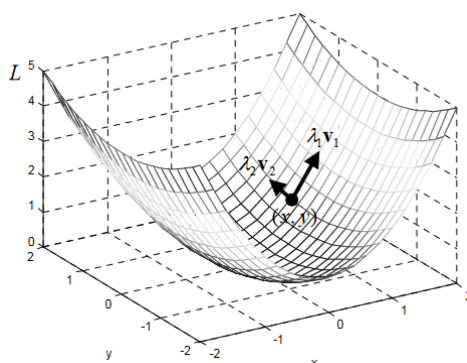


Figure 2.15: Principal Curvature and Direction(Choi et al., 2009).

According to the author's experimental testing, principal curvature achieves EER = 0.018% which is quite optimal solution. As compared with the maximum curvature algorithm, the datasets are of lower quality, but the author's method still performed better than the novel methods. It shows that the veins were identified in continuity form as compared with the other methods. The author applied the maximum curvature algorithm on his in-house datasets, the extracted veins were extracted not as good as mentioned in the original author's literature. This is probably due to the principal curvature methods performing gradient normalization which reduces the effect of uneven brightness on the feature extraction algorithm. Maximum curvature doesn't consider this factor and hence results in worse performance.



Figure 2.16: Finger Veins Pattern Extracted Using Principal Curvature (Choi et al., 2009).

2.4.4 Wide Line Detector

The concept of local maximum and minimum of the images tends to identify the center of the veins, but the width of the veins could be lost during the feature extraction. Principal curvature identifies the eigenvalues and eigenvectors from the Hessian Matrix and is sensitive to the noise pixels due to the nature of the second-order partial derivative. The wide line detector algorithm aimed to extract all the pixel values on the lines which recovers the thickness of the veins as much as possible (Liu, Zhang and You, 2007). It does not use any derivative method, but a circular filter is used instead to cover different widths of the lines. Huang et al. (2010) applied the wide line detector method to extract the finger veins. The extracted features were performed self-proposed normalization to ensure that the templates from the database image and processed image had a similar orientation. The normalization is useful for biometrics cases, but it is not required for the medical vein viewer application.

The size of the finger vein images was reduced firstly to reduce the computational cost and time. The size-reduced images were performed wide line detection to identify the veins pattern. In the first step, the *weighted mask having similar brightness* (WMSB) score is calculated by detecting the group of pixels within the mask which has a brightness level like the center of the mask.

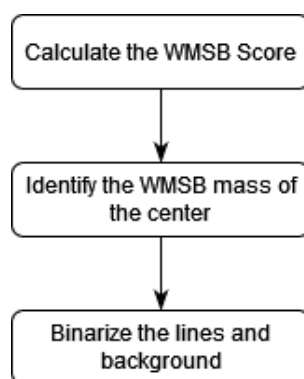


Figure 2.17: Flowchart of Wide Line Detector Algorithm (Huang et al., 2010).

The circular mask as shown in Figure 2.17 is located on the dark line (veins) where the center of the mask has the darkest brightness. Most of the pixels in the mask have a brightness like the center and therefore the WMSB score is low. If the mask is fully located on the white background, the WMSB

score would be at the global maximum. Liu, Zhang and You found out that the threshold, t could be set as the standard deviation of the input image and rounded to the nearest integer through their experiment.

$$s(x, y, x_0, y_0, t) = \begin{cases} 0, & F(x, y) - F(x_0, y_0) > t \\ 1, & \text{otherwise} \end{cases} \quad (2.6)$$

where

$s(x, y, x_0, y_0, t)$ = WMSB score

(x_0, y_0) = coordinate of the center of the mask

(x, y) = coordinate of the pixels in the mask except the center

t = brightness contrast threshold

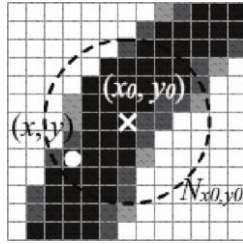


Figure 2.18: Circular Mask on the Finger Veins Image (Huang et al., 2010).

Then, the WMSB mass of the centre is calculated by finding the WMSB score for all pixels within the circular mask.

$$m(x_0, y_0) = \sum_{(x,y) \in N(x_0,y_0)} s(x, y, x_0, y_0, t) \quad (2.7)$$

where

$m(x_0, y_0)$ = WMSB mass of the centre

$N(x_0, y_0)$ = Neighbourhood region of the mass of the centre

Finally, the feature image is obtained by generating the binary image of the veins pattern (white pixel) and background (black pixel). If the calculated mass of the centre is above the threshold, g , it is identified as background while lower than that is the veins. Liu, Zhang and You highlight that the threshold, g was set as $\pi r^2/2$ in their proposed algorithm.

$$V(x_0, y_0) = \begin{cases} 0, & m(x_0, y_0) > g \\ 255, & otherwise \end{cases} \quad (2.8)$$

where

$V(x_0, y_0)$ = Feature image

g = geometric threshold

About the guideline of creating a circular mask, Liu, Zhang and You found out that the algorithm was optimized when the radius of the mask $\geq 2.5 \times$ widths of lines detected. However, due to the veins' widths are not the same in the image, this relationship could only act as a guideline to search for a suitable parameter value. An unoptimized parameter chosen would slow down the line detection process or unable detect the lines, but one thing is certain the circle mask must be larger than the line width.

The author assessed the output of the proposed algorithm by performing template matching between two images taken from different sessions. The algorithm performed excellently with EER = 0.87% using low-quality datasets. Although the algorithm seems to identify most of the dark pixels that are likely veins, there are more noticeable noises appear in the image. This is probably due to lack of pre-processing steps in the algorithm which cause the feature extraction algorithm identify noise pixels as dark lines. Secondly, the brightness of the image is uneven due to the image captured conditions. This is a common problem due to the uneven lighting and light distraction from the environment.

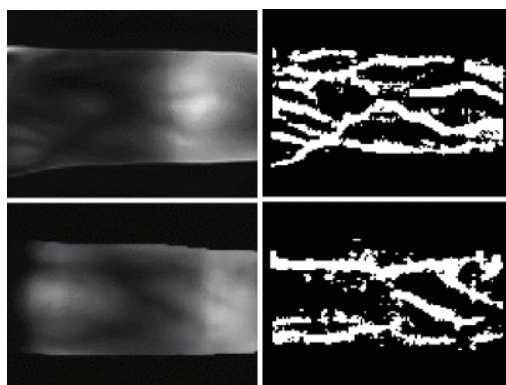


Figure 2.19: Finger Veins Pattern Extracted Using Wide Line Detector (Huang et al., 2010).

2.4.5 U-Net Convolutional Neural Network

U-net is one of the architectures of convolutional neural network (CNN) which was proposed for biomedical applications as well (Ronneberger, Fischer and Brox, 2015). It solves the major issue in biomedical imaging which is the scarcity of datasets. The author applied the data augmentation technique to enlarge the small number of datasets such as distortion, deformations, rotation, etc which achieve intersection over union (IoU) 92%. During the early learning stage of the CNN model, it is still learning to identify the basic features such as lines, curves, circles, etc. Data augmentation concept is by letting the model learn about basic features through the same images but looks differently.

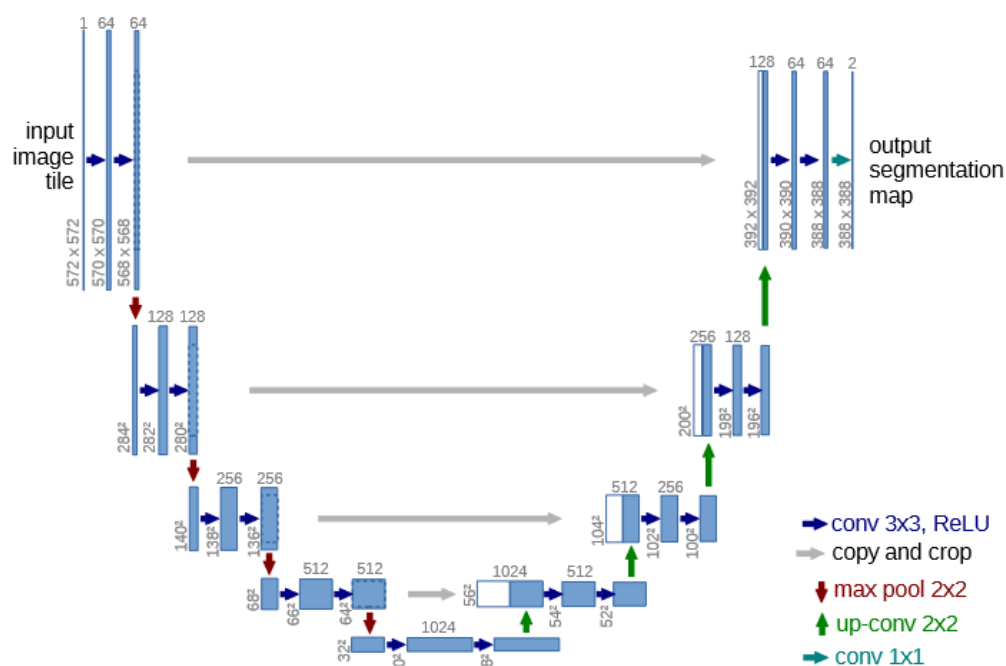


Figure 2.20: U-Net Architecture of a CNN model (Ronneberger, Fischer and Brox, 2015).

There are 4 operations throughout the U-Net training process. The first one is 3×3 convolutions with ReLU activation function. This layer is aimed to reduce the size of the image by 2 pixels on each side and train to learn about the important information from the data. The second operation which is max pooling 2×2 with a stride of 2 reduces the size of the x-y feature map. Max-pooling is used to minimise the dimensionality of the features to

remove densely weighted features that are easier to calculate and consume fewer computer resources for parameter learning. In other words, the image remains with important information, but the spatial information is removed. Next, the 2×2 up convolutions transfer the important information to a higher resolution image by scaling each vector by 2. At this stage, the image consists of spatial and graphical information. Last but not least, the 1×1 convolution operation generates the segmentation results which consist of two channels which are foreground and background. The output image should be a smaller size than the input image but it's still consisting of spatial and graphical information in the segmentation results.

Marattukalam and Abdulla (2020) first proposed NIR veins segmentation using the U-Net framework on the palmer datasets. The author modified the first block by using the Gabor filter which is a Gaussian kernel function. The training images were greyscale images which were processed to become binary image which contains veins as the foreground. The pre-processing techniques applied include ROI extraction, adaptive thresholding, morphological opening operation, and Medial Axis Thinning algorithm (Lee, Kashyap and Chu, 1994). Binary masks are used for training data of the U-Net model and as ground truth to compare the prediction results from the trained model. As there was no manually labelled ground truth image available, the evaluation of the model's performance was carried out using the Dice Coefficient and Jaccard Index metrics. The datasets applied were only 6000 images without mentioning data augmentation was performed. Hence, the performance of the model could be enhanced by using more training datasets or implementing data augmentation.

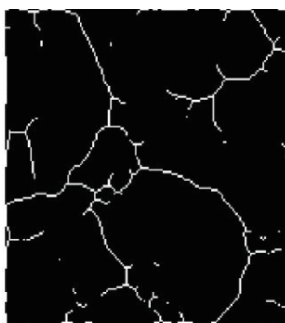


Figure 2.21: Morphological and Skeletonization Veins Mask (Marattukalam and Abdulla, 2020).

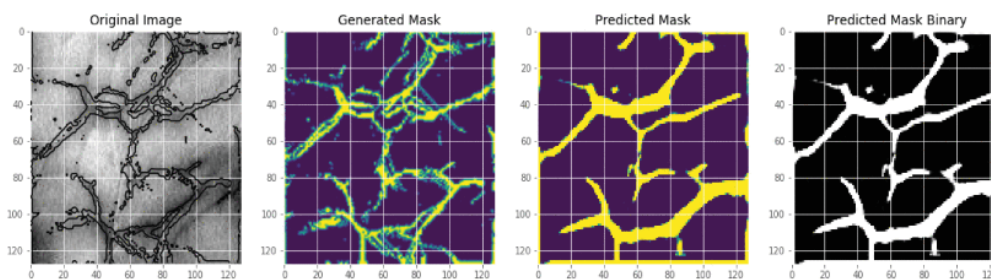


Figure 2.22: Veins Segmentation Results from U-Net (Marattukalam and Abdulla, 2020).

Jing et al. (2021) performed forearm veins segmentation using a modified U-Net model. The author used the in-house datasets which were small in size to directly train the model. The data augmentation technique used such as shifting, rotation, flipping, etc to generate more images to train a robust model. The augmented datasets were further split into training and validation purposes to evaluate the model trained while tuning the hyperparameters. Besides using the 2×2 Upsampling convolution in the expansion path, it was found that 3×3 Conv2DTranspose layers could reduce the probability of noises being identified as veins. The batch normalization layers were used to reduce the problem of updating coordinates in different layers. The batch normalization layer was applied to standardize the data, where the mean was set to 0 and the standard deviation was set to 1. The training process would be more stable and faster due to the weightage of each layer does not change drastically. The novel method achieved high accuracy (0.9909) and specificity (0.997) score. However, the recall (0.4856) of the model was lower than half which means that more than half of the vein's pixels were not detected. The recall and precision metrics still have room to be improved to avoid overfitting. The large size of the datasets could be used and performed hyperparameter tuning to optimize the trained model.

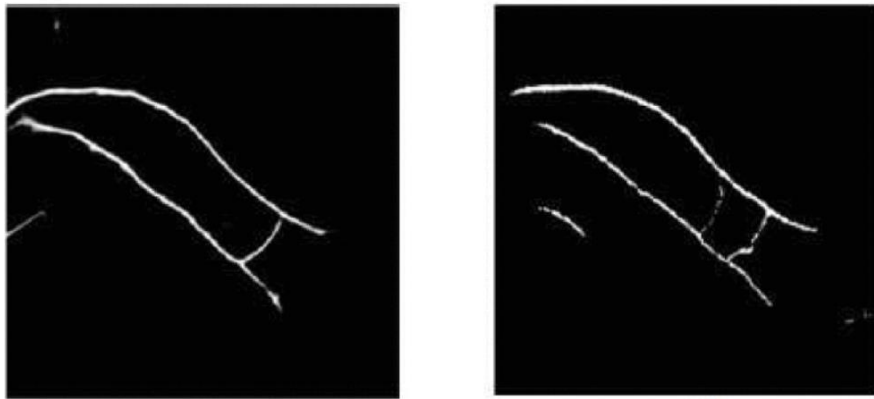


Figure 2.23: U-Net Segmented Veins (Left) and Ground Truth (Right) (Jing et al., 2021).

Table 2.3: Veins Pattern Segmentation Comparison of U-Net Model.

	Marattukalam and Abdulla (2020)	Jing et al. (2021)
Modification from original U-Net	Replaced Conv 3×3 with Gabor Filter	Replaced 2×2 Upsampling Conv with 3×3 Conv2DTranspose and added batch normalization layers
Datasets	6000 palmer datasets	> 20,000 augmented forearm images
Learning Rate	0.00015	0.0001
Activation Function	-	ReLU
Number of Epochs	20, 30, 40, and 50	25
Performance	Dice Coefficient = 0.69 Jaccard Index = 0.71	Specificity = 0.997 Accuracy = 0.991 Recall = 0.486 Precision = 0.660

2.5 Transfer Learning

Transfer learning is put into practice to improve the learning progress of a new task by utilizing knowledge obtained from a learned task that was previously

known. There were several authors had proposed transfer learning on veins feature extraction tasks.

Fairuz, Habaebi and Elsheikh (2018) applied transfer learning technique on a pre-trained AlexNet CNN model to perform finger vein identification. Three experiments were conducted by varying the input images numbers, mini batch size, and number of epochs. The most optimal result was obtained in the experiment with 400 input images, a batch size of 50, and 10 epochs, resulting in an accuracy of 95% and an area under the curve (AUC) of 0.99.

Kuzu, Maiorana and Campisi (2020) proposed a custom network for vein feature extraction by modifying the DenseNet161 network. The custom embedder was added just before the classification layer and consisted of global average pooling, batch normalization, dropout (50%), a fully connected layer, and batch normalization. The network weights were initialized from ImageNet, and it was found that this outperformed the weights initialized by default. The author also performed open-set and closed-set scenario training, and ultimately, the closed-set scenario performed the best. The modified DenseNet161 with ImageNet weight initialization achieved the best results when tested on three different datasets. The SDUMLA (finger vein) dataset achieved an EER of 0.037%, the Poly U-P (multispectral palmprint) dataset achieved an EER of 0.005%, and the Bosphorus (dorsal vein) dataset achieved an EER of 0.7%. The findings of this study demonstrate that modifying a pre-existing custom network and utilizing transfer learning can yield superior model performance, rather than creating a new network from the ground up.

(Garcia-Martin and Sanchez-Reillo, 2021) had proposed a transfer learning method on vein feature extraction for biometric recognition using a smartphone. The study utilized four different pre-trained networks, namely VGG16, VGG19, ResNet50, and ResNet152. The final fully connected layers were replaced with simplified fully connected layers that were randomly initialized. To prevent the pre-trained weights from being altered during backpropagation, the parts before the fully connected layers were frozen. The body was only unfrozen once the head had learned the critical features of the dataset. At that point, the body was unfrozen, and the learning rate was reduced. The four pre-trained networks were used as feature extractors and

applied to three original datasets which are PUT, UC3M-CV2, and UC3M-CV1. The results were evaluated in terms of EER, with values of 0.38%, 0.78%, and 2.04% achieved on the respective datasets. Overall, the study demonstrates that transfer learning can be a useful approach for vein feature extraction in biometric recognition, and pre-trained networks such as VGG16, VGG19, ResNet50, and ResNet152 can be effective feature extractors when appropriately modified and trained.

2.6 Pre-trained Models

2.6.1 ResNet

ResNet (Jian, 2015) is a neural network architecture that was developed in 2015 by researchers at Microsoft Research Asia. It is specifically designed to tackle the issue of vanishing gradients that occurs in very deep neural networks. In a typical neural network, each layer transforms the input and passes it as the input to the next layer, with the weights of each layer adjusted during training to minimize the error between the ground truth and predicted output. However, in very deep neural networks, the gradients can become extremely small, making it difficult for the network to learn anything further. ResNet overcomes this challenge by introducing shortcut connections that allow the input to bypass one or more layers and be added to the output of later layers. These connections facilitate the flow of information through the network, making it easier for gradients to propagate and enabling the network to learn deeper and more complex representations. By using these connections, ResNet focuses on the essential features of the input, preventing the network from becoming overwhelmed by the noises.

2.6.2 ResNeXt

ResNeXt (Xie et al., 2017) is a deep neural network architecture presented by the Microsoft Research team in 2017, which addresses the challenge of scaling and generalization in deep learning models. ResNeXt aims to enhance the neural networks' performance by increasing their capacity while preventing overfitting. In contrast to conventional neural networks, ResNeXt employs a novel building block called the "cardinality bottleneck," which splits channels into several groups and processes them in parallel. This bottleneck expands the

network's capacity while avoiding a rise of the quantity of parameters, which can lead to overfitting. ResNeXt also utilizes a hierarchical feature learning approach, where small convolutional filters initially process the input, followed by larger ones. This approach allows the network to learn different features at various scales, resulting in improved performance on a range of tasks. The author also claimed that ResNeXt outperformed ResNet family and Inception family models. Therefore, ResNeXt provides a more effective and efficient way to scale neural networks while maintaining their generalization ability.

2.6.3 VGG

The VGG (Simonyan and Zisserman, 2015) architecture was introduced in 2014 by researchers from the University of Oxford to address the challenge of image recognition in deep learning. The network achieves this goal by significantly increasing its depth while maintaining a fixed convolution kernel size of 3x3. This increase in depth enhances the network's ability to recognize and classify objects in images. To prevent overfitting, VGG employs a novel regularization approach that stacks small convolutional filters followed by pooling layers, enabling the network to learn a hierarchy of increasingly complex features that is crucial for image recognition.

VGG has several variations, including VGG16 and VGG19, 'which have varying amount of convolutional layers. The VGG16 architecture comprises 13 convolutional layers, in contrast of VGG19 has 16 convolutional layers, but both also consists an additional of 3 fully connected layers. Overall, VGG represents a significant departure from existing deep learning models by increasing the depth of the network and using small convolutional filters to improve image recognition. This approach also enhances the network's ability to resist overfitting, making VGG an essential benchmark for image recognition tasks and a widely used architecture in various applications.

2.6.4 Inception

The Inception (Szegedy et al., 2014) neural network architecture was originated by researchers from Google in 2014 to improve the efficiency and

accuracy of image classification. Its main goal was to minimize computational cost and memory usage while enhancing neural network performance. Unlike traditional neural networks, which rely on manual selection of filter size and number, Inception employs an "Inception module," which allows the network to learn the optimal filter size and number for each layer. This module combines multiple convolutional filters and pooling operations that are applied in parallel to the input. The next layer receives the concatenated outputs as input. Inception also uses "dimensionality reduction" to decrease the network's parameter quantity, allowing it to capture essential features while reducing computational cost and memory usage. The architecture has different variations such as Inception v1, v2, and v3. These innovations in Inception architecture helped to enhance the capabilities of neural networks while minimizing the computational cost and memory usage, making it a widely used benchmark for image classification tasks.

2.6.5 DenseNet

DenseNet (Huang et al., 2016) is an architecture of deep neural network that was proposed by the team from Facebook AI Research in 2017 to address the issue of vanishing gradients that occurs in traditional deep neural networks. The goal of DenseNet is to enhance the flow of information and gradients in the network while keeping the number of parameters at a minimum. DenseNet was introduced to tackle the vanishing gradient problem by introducing a new building block known as the "Dense block," which enables the network to reuse feature maps from previous layers. In a Dense block, the output of each layer is concatenated with the outputs of all preceding layers and fed as input to all subsequent layers. This results in dense connections between all layers, facilitating the flow of information and gradients throughout the network.

The DenseNet architecture also employs a technique called "bottleneck layers," which compress the feature maps using 1x1 convolutions before applying 3x3 convolutions, minimizing the possible amount of parameters in the network while preserving its essential features. DenseNet has several variations, including DenseNet-121, DenseNet-169, and DenseNet-201, with varying numbers of layers and parameters. DenseNet-121, for

example, comprises four dense blocks and has 7.98 million parameters, whereas DenseNet-201 has six dense blocks and 18.3 million parameters.

2.6.6 SENet

SENet (Hu et al., 2017), or Squeeze-and-Excitation Network, is an innovative deep neural network architecture that aims to address the limitations of the traditional CNN in modeling interdependencies between channels. The researchers from Huazhong University of Science and Technology introduced SENet in 2017 with the primary settings of enhancing a neural network's ability to learn feature dependencies and relationships across channels.

While traditional CNNs learn features by convolving input images with filters, their inability to effectively model channel dependencies can impact their performance. To resolve this, SENet introduces the "Squeeze-and-Excitation block," a new building block that recalibrates the feature maps based on interdependencies between channels. The block comprises a "squeeze" operation to reduce the spatial dimensions of feature maps, followed by an "excitation" operation that applies a set of learned weights to each channel independently.

By learning to focus on the most informative channels and suppress the less informative ones, the excitation operation improves feature representation and classification accuracy. This block can be easily integrated into existing CNN architectures. SENet achieved state-of-the-art results on the ImageNet dataset in 2017 with SENet-154, a 154-layer variation of the network. SENet has also been extended beyond image classification to tasks like object detection, semantic segmentation, and video classification.

2.6.7 MobileNet

The next architecture introduced is MobileNet which was developed by researchers from Google in 2017 to enable deep neural networks to be operated on mobile devices that have constrained computational capabilities. The primary aim of MobileNet is to provide a lightweight architecture with high accuracy for tasks related to the classification of images while minimizing the number of parameters and computational cost.

Traditional deep neural networks have a significant amount of parameters and need significant computing power, making their deployment impractical on mobile devices. To address this issue, MobileNet introduces a new building block called the "depthwise separable convolution." Its purpose is to split the standard convolution operation into two separate operations: utilizes a type of convolution called depthwise convolution, that involves the application of a single filter to every input channel independently, and a pointwise convolution that applies a 1x1 filter to merge the output of the depthwise convolution. This technique reduces the amount of parameters and computational cost while maintaining high accuracy on image classification tasks.

MobileNet also uses a technique called "width multiplier" that reduces the number of channels in the network by a factor of the multiplier, which further reduces the amount of parameters and computational cost without significantly impacting accuracy. MobileNet has several variations, including MobileNetV1, MobileNetV2, and MobileNetV3, which differ in the number of layers and the use of different techniques to improve accuracy and reduce computational cost.

2.6.8 EfficientNet

In 2019, a group of Google researchers introduced EfficientNet (Tan and Le, 2019a) which is a neural network architecture to tackle the challenge of achieving high accuracy on image classification tasks while minimizing the number of parameters and computational cost. Unlike existing models, which optimize for either depth, width, or resolution, EfficientNet employs a technique called compound scaling that simultaneously optimizes all three factors. This enables it to achieve state-of-the-art accuracy with fewer parameters and lower computational cost.

To achieve this, EfficientNet scales network depth, width, and resolution simultaneously using a fixed scaling coefficient, which is optimized using a grid search algorithm to minimize the validation loss. Additionally, it introduces a new building block called the "MBConv block," which is a combination of depthwise and pointwise convolutions with an inverted bottleneck structure that improves feature representation.

EfficientNet has variations like EfficientNet-B0 to B7, which differ in the number of layers and scaling coefficients used in the compound scaling method. It has demonstrated exceptional results in several computer vision tasks, which included semantic segmentation.

2.7 Summary

During the normal practice of intravenous cannulation, medical practitioners would consider median cubital superficial veins located at the cubital fossa. However, it is often that the veins are not visible to be seen using the traditional method. NIR imaging can capture the veins image using two approaches, which are reflection and penetration. There are two groups of methods to extract the features from the NIR image, which are computer vision techniques and deep learning techniques.

Computer vision techniques are designed based on the geometrical structure of the veins and come with a complex handcrafted feature extraction which would bring out several problems as below (Qin et al., 2019).

- i) The assumptions are not constantly successful in extracting vein patterns.
- ii) It is difficult to interpret the characteristics of all patterns generated by pixels.
- iii) It is challenging to construct a mathematical model that can accurately represent patterns such as valleys or lines.

However, computer vision techniques can be easily implemented without the actual need to understand the mathematical principle behind the algorithm if the source code was opened to the public. On the other hand, deep learning techniques don't require searching for patterns from the image, but the model will do so. The implementing process is focused on optimizing the parameters and layers of model architecture which could be time-consuming for the trial-and-error process. The advantage is that deep learning techniques could achieve higher accuracy of the veins segmentation easily compare with computer vision techniques. In contrast, transfer learning enable to develop a deep learning by using small amount of dataset. Fine-tuning on pre-trained

model was done to address new tasks and the training time is largely reduced as those generic features has been learned.

Based on the literature study, the performance of their vein's segmentation is summarized in Table 2.4. Due to the performance metrics were not standardized for all algorithms, it would not be fair to compare which algorithm performed better based on their self-published data. Therefore, the performance of some literature algorithms would be evaluated in the results section.

Table 2.4: Summary of Performance of Literature Method.

Veins Segmentation Algorithm	Parts of the body	Performance
Repeated Line Tracking	Fingers	EER = 0.145%
Maximum Curvature	Fingers	EER = 0.0009%
Principal Curvature	Fingers	EER = 0.018%
Wide Line Detector	Fingers	EER = 0.87%
U-Net CNN (Marattukalam and Abdulla)	Palmer	Dice Coefficient = 0.69 Jaccard Index = 0.71
U-Net CNN (Jing et al)	Forearm	Specificity = 0.997 Accuracy = 0.991 Recall = 0.486 Precision = 0.660
AlexNet Transfer Learning (Fairuz, Habaebi and Elsheikh)	Fingers	Accuracy = 95% AUC = 0.99
DenseNet161 Transfer Learning (Kuzu, Maiorana and Campisi)	Fingers Multispectral palmprint Dorsal	EER = 0.037% EER = 0.005% EER = 0.7%
VGG16, VGG19, ResNet50, and ResNet152 Transfer Learning (Garcia-Martin and Sanchez-Reillo)	Wrist	EER = 0.38% (UC3M-CV2) EER = 0.78% (UC3M-CV1) EER = 2.04% (PUT)

CHAPTER 3

METHODOLOGY AND WORK PLAN

3.1 Introduction

In this study, a well-performed vein feature extraction algorithm for medical applications was proposed. Due to the complexity of the image acquisition process involving hardware construction and compliance with code of standards, obtaining forearm vein datasets was deemed too time-consuming for the study's duration. Therefore, NIR datasets were obtained from the authors through their published journal papers. The datasets used for this study included images of the arm and palm of the upper limb, which were classified based on the appropriate method for extracting the region of interest (ROI) since the datasets included objects other than the forearm. Both automatic and manual ROI extraction were carried out in this study.

The images obtained from the NIR camera had low quality, which requires the application of pre-processing techniques to improve the images. Issues such as uneven illumination brightness and noise were prevalent due to camera limitations. The veins pattern was then extracted using state-of-the-art techniques and self proposed method as discussed in CHAPTER 2, and the performance of several feature extraction methods was compared using common evaluation metrics utilized in the segmentation problem.

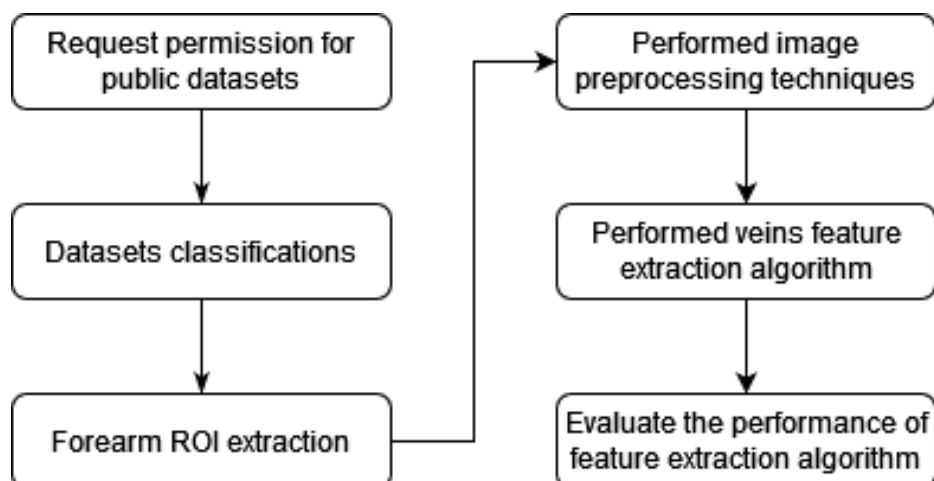


Figure 3.1: Process Flow of the Study.

In this study, the hardware specifications used for image processing and algorithm development were an Intel I5 processor @ 2.3 GHz, an Nvidia graphics card GTX1060, 8 GB of RAM, and a Windows 10 operating system environment. MATLAB R2020a were used to run the state-of-the-art algorithms while the proposed models were carried out on Microsoft Visual Studio Code using Python 3.9.15 and TensorFlow framework (2.10.1).

3.2 Datasets Handling Process

Among the three publicly available datasets as shown in Table 2.2, there are only two consists of ground truth datasets which are NTUIFDB v1 and VeinCV_RL. Although NTUIFDB v2 is the largest public dataset found, it is not considered in this study as there were no ground truth datasets prepared by the author. NTUIFDB v1 would be the suitable dataset for this study where it consists of 250 forearms NIR images with its ground truth data to test the performance of the feature extraction algorithm.

The NTUIFDB v1 dataset was collected from 110 individuals from 9 countries and was introduced in a research paper in 2012 (Zhang et al., 2012). The dataset consists of images captured in two sessions, with varying light intensities and angles. The position of the subjects' arm was not standardized in the two sessions. The dataset has various issues, which were divided into five groups to apply different image preprocessing methods. The characteristics of these five groups of images are summarized below:

- i) 2 rulers on the left and right side of the image and 2-floor tile lines above and below the forearm.
- ii) Same characteristic as (i) but lack of 1 ruler or floor tile elements
- iii) 2 rulers on the left and right side of the image and a wide ruler below the forearm
- iv) Same characteristic as (i) but with forearm shadow
- v) Same characteristic as (iv) but forearm position touches or falls outside the rulers and floor tile lines

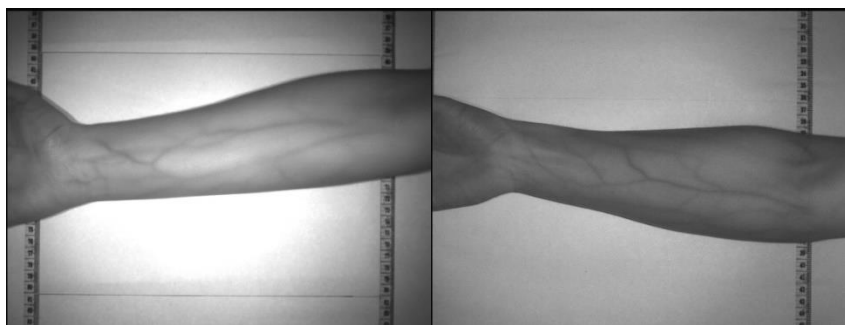


Figure 3.2: Group I (on left) and Group II (on right) of NTUIFDB v1.

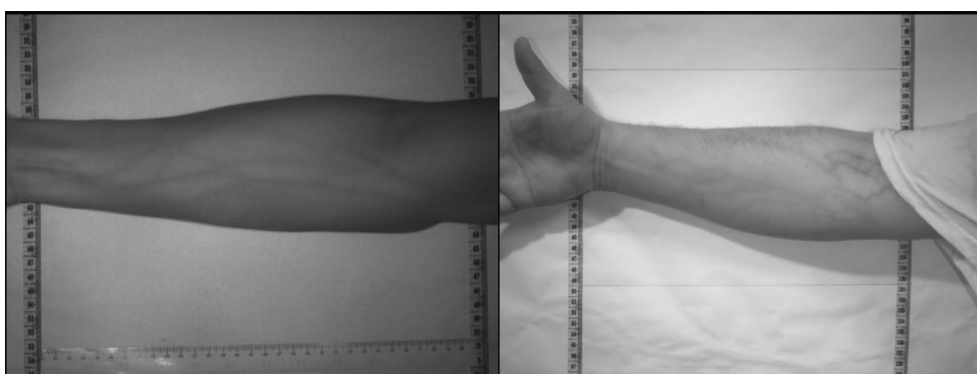


Figure 3.3: Group III (on left) and Group IV (on right) of NTUIFDB v1.

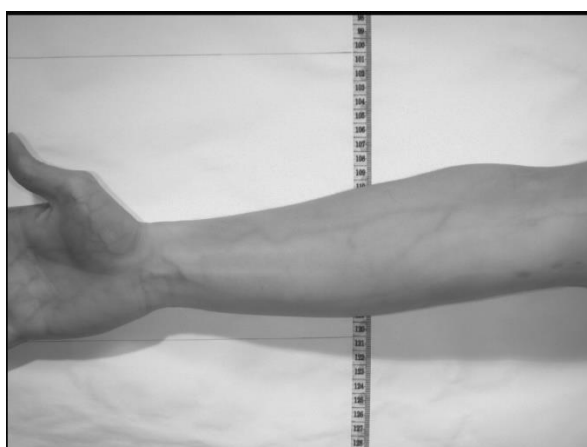


Figure 3.4: Group V of NTUIFDB v1.

Among all the groups, datasets from Group V had the most critical issues as shown in Figure 3.4. The forearm laid on top of the ruler while the ruler was in the middle of the forearm. The palmer side of the hand was included in the image where no visible lines to divide the palmer and forearm parts. Besides that, there were some traits found on the forearm such as tattoos,

lush hair, and sleeve of their clothes on the minor subjects across different groups.

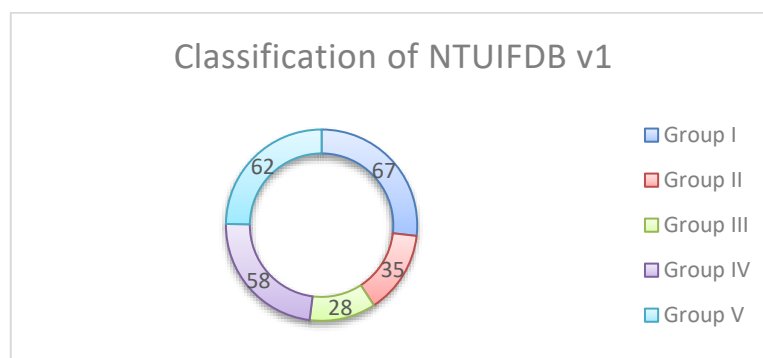


Figure 3.5: Distribution of NTUIFDB v1 Datasets.

3.3 Forearm ROI Extraction

3.3.1 Automatic ROI Extraction

The Group I image of NTUIFDB v1 has the most complete guideline to segment the forearm region. The 2 rulers located on the left and right side of the forearm formed a closed boundary box with the 2 horizontal floor tile lines. The region inside the box is exactly the forearm region to be extracted.

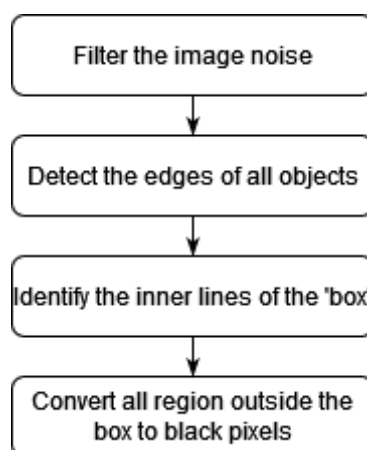


Figure 3.6: Flowchart of Forearm Edge Detection.

Firstly, the raw input image consists of digital image noises. It would identify these noises as an object if edge detection was straight away performed. These grain-like noises were probably caused by the high ISO value of the camera (Marie Gardiner, 2020). It amplifies the brightness of the

image, but the side effect is less cleaned image would form. Using adaptive filtering would eliminate the grains of the image and return a cleaner image. Then, edge detection could be performed to highlight the edges of all objects. Canny edge detection is utilized to identify all edges present in the images. Second-order derivative edge detection is very sensitive to all edges which include minor noises. It was found that applying the [0 0.1] vector threshold would be sufficient to detect large objects.

To identify the vertical rulers at two sides of the forearm and the horizontal floor tile lines, Sobel edge detection was used to detect the vertical and horizontal edges respectively as shown in Figure 3.7. Both rulers and floor tiles have a strong detection result which can be considered as long lines. These four long lines intersect each other and form a 'box', where the forearm is within the 'box'. Pixels which are located outside the 'box' would then be converted to black pixels and the forearm would retain its pixel information. During the canny edge detection, some forearm edges were discontinued. The gap was filled by performing a morphological closing operation using a line as a structural element. From the experience of trial and error, 5 pixels of the line can connect the gap of edges without connecting other unwanted lines.

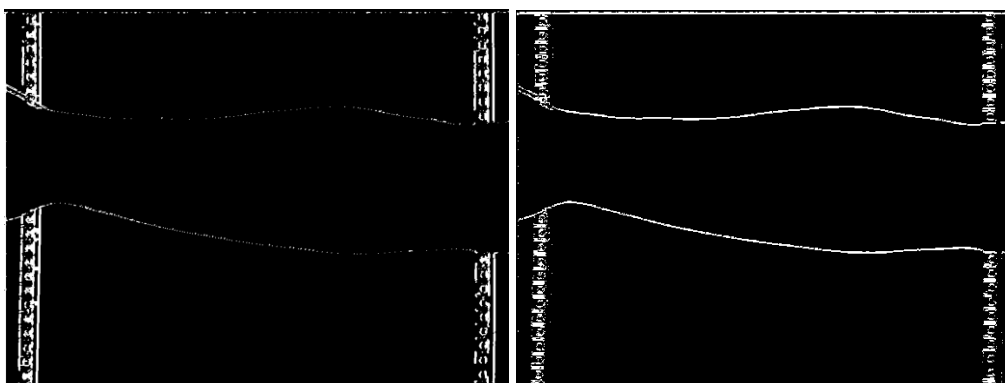


Figure 3.7: Sobel Edge Detection on Direction: Vertical (left) and Horizontal (Right).

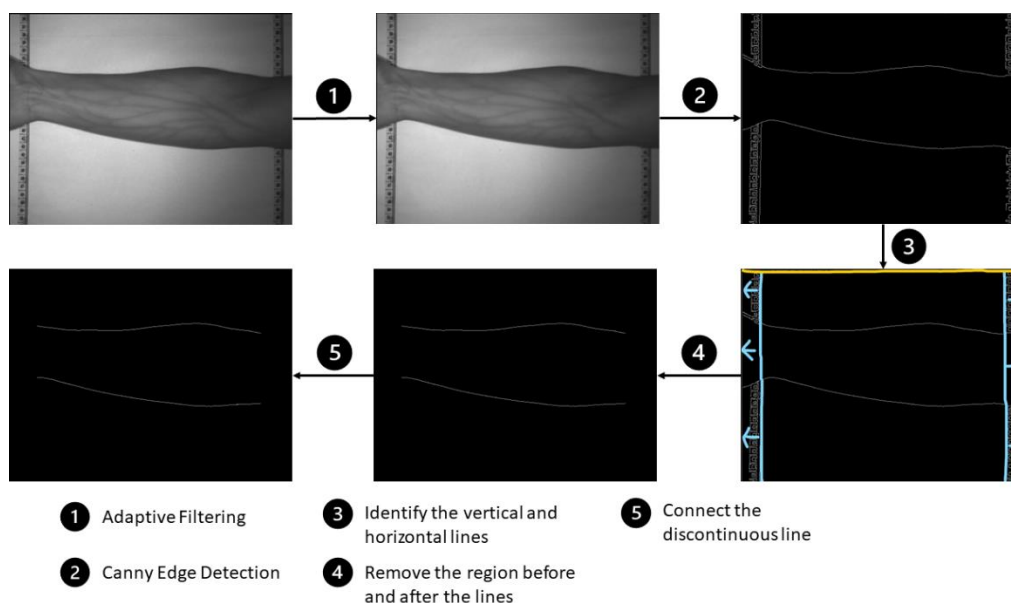


Figure 3.8: Visualizing the Forearm Edge Detection Process.

The above-explained methods were suitable to be used in Group I and IV datasets. For the Group II ROI extraction algorithm, the only difference is during the detection of the horizontal edges. The detection of horizontal edges would only stop when there are two horizontal edges detected in Groups I and IV. The same methods used in Group III would cause the forearm edge identified as the boundary of the ‘box’. Therefore, when defining the number of pixels detected would be considered as floor tile line, it should be set between the forearm edge and floor tile edge as illustrated in Figure 3.9. It ensures the algorithm would not detect the forearm edge as a floor tile edge when the ‘real’ floor tile edge is absent.

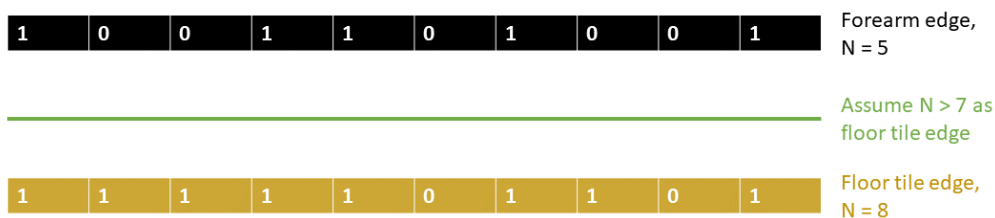


Figure 3.9: Illustration of Defining Floor Tile Edge.

During searching for the two horizontal lines in Groups I and IV, the searching starts from the first and last row of the image towards the center. Since there are two lines in the image, during defining the number of pixels as

floor tile edge, the value can be set lower than the forearm edge as well. It ensures that the unclear floor tile lines would also be detected as well. This method requires assuming the width of the floor tile so that it can be fully removed from the image. In Group III, the assumptions of the width of horizontal lines should be larger than in other groups. The trial-and-error results showed that the width = 80 is optimized in Group III while width = 30 is sufficient for other groups.

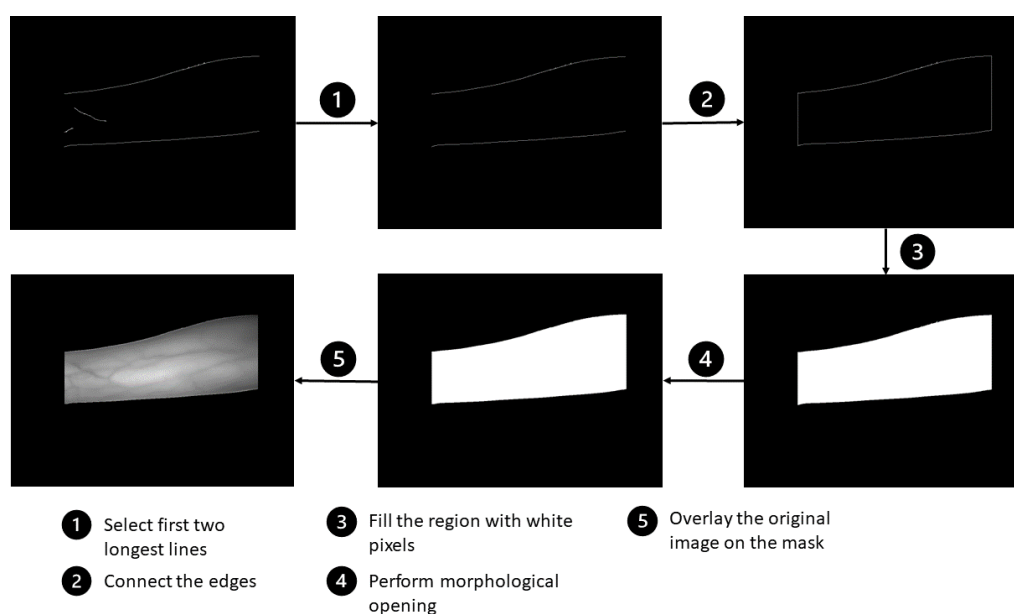


Figure 3.10: Visualizing the Process of ROI Extraction Using Binary Mask.

When performing the edge detection in the above-mentioned process, there were other edges besides the forearm edges being detected. It can be observed that in most cases, forearm edges were the longest edge being detected. Therefore, the first two longest lines remain while another line would be converted to black pixels. The two lines were further connected between the two endpoints of the lines to form a closed contour. It can be then identified as a shape and filled with the pixels within the closed contour with logic '1'. A mask is then obtained which represents the ROI of the forearm in binary form. The ROI extraction of the forearm raw image can be done by overlaying it on the mask.

The above-mentioned process could be done in the same way in Group I to III datasets. However, the edges of shadow in the Group IV datasets

would be detected and the lines are often longer than the forearm edges. Therefore, the elimination of the shadow edge should be done before continuing the algorithm. This can be done by searching for the lines nearest to the lower left corner and the total pixels of the line should be more than the threshold set. This is to avoid those thin small lines being detected. Once the shadow edge is removed, the same ROI extraction steps can be done together with the other groups.

3.3.2 Manual ROI Extraction

Group V datasets were unable to perform automatic ROI extraction as discussed in Section 3.3.1. The automatic thresholding algorithm is designed based on the assumption of the forearm within the ‘box’. If the forearm region were touching the ruler edges or partially outside, the ruler edges don’t assist in segmenting the forearm, wrist, and cubital fossa. Therefore, a manual approach to perform ROI extraction on these datasets was proposed using Computer Vision Annotation Tool (CVAT).

The advantage of using CVAT to manually segment the ROI of the forearm is the built-in AI interactor, Deep Extreme Cut (DEXTR). Selecting the four corners of the forearm could identify the forearm region in an accurate manner. With the help of DEXTR, the segmentation process was much shorter as compared with other segmentation tools.

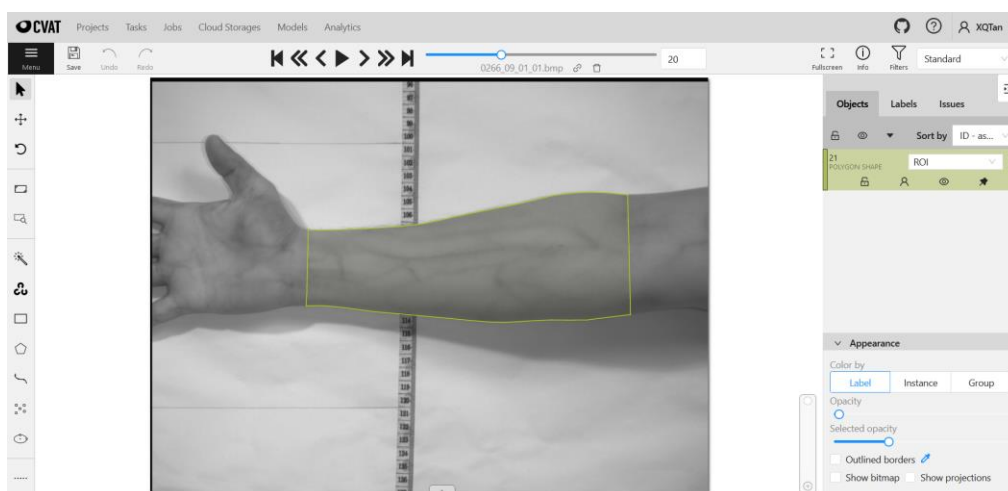


Figure 3.11: Manual Forearm ROI Extraction on CVAT.

3.4 Image Preprocessing

There are two issues can be observed from the raw images, which are uneven illuminance of the image and digital image noise. The first problem causes the brightness to be unevenly distributed across the forearm. The center of the image would appear in higher brightness while the boundary of the forearm is much dimmer. This can be solved by applying the contrast limited adaptive histogram equalization (CLAHE) that can evenly distribute the brightness of the image. When the operation stretches the histogram distribution of the image, it limits the contrast level to avoid more image noise being introduced.

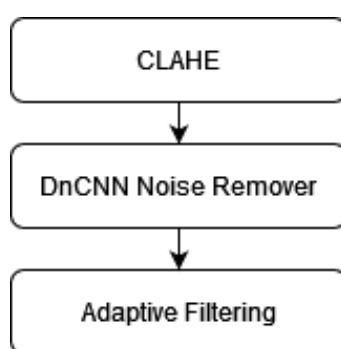


Figure 3.12: Flowchart of Image Pre-processing.

Besides that, the image enhanced using CLAHE made the noises of the image more visible to be seen. By applying both Denoising Convolutional Neural Network (DnCNN) and adaptive filtering could largely improve the noisy condition of the image. DnCNN is one of the convolutional neural network that can denoise the image generally through the residual learning strategy (Zhang et al., 2017). It learns about the noise info of the input image at the hidden layer and generates a similar noise layer. The denoising of the image could be achieved ideally by subtracting the noise layer from the image. Below is the architecture of the DnCNN applied for image denoising.

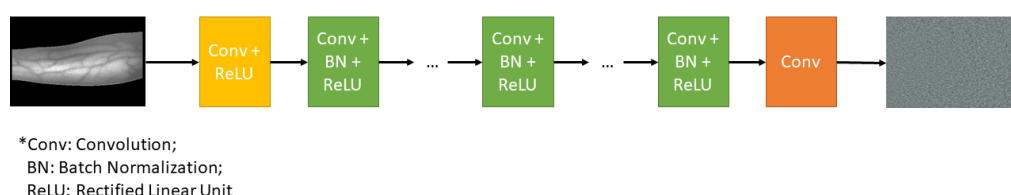


Figure 3.13: Architecture of DeCNN.

After performing DnCNN, the large texture of grain-like noises was removed. The surface of the output image was still unevenly distributed. Adaptive thresholding could remove these noises while at the same time fixing the discontinuous line and smoothing the image (Knutsson, Westin and Kikinis, 2000). After performing three steps of pre-processing techniques, the veins pattern was ready to be extracted.

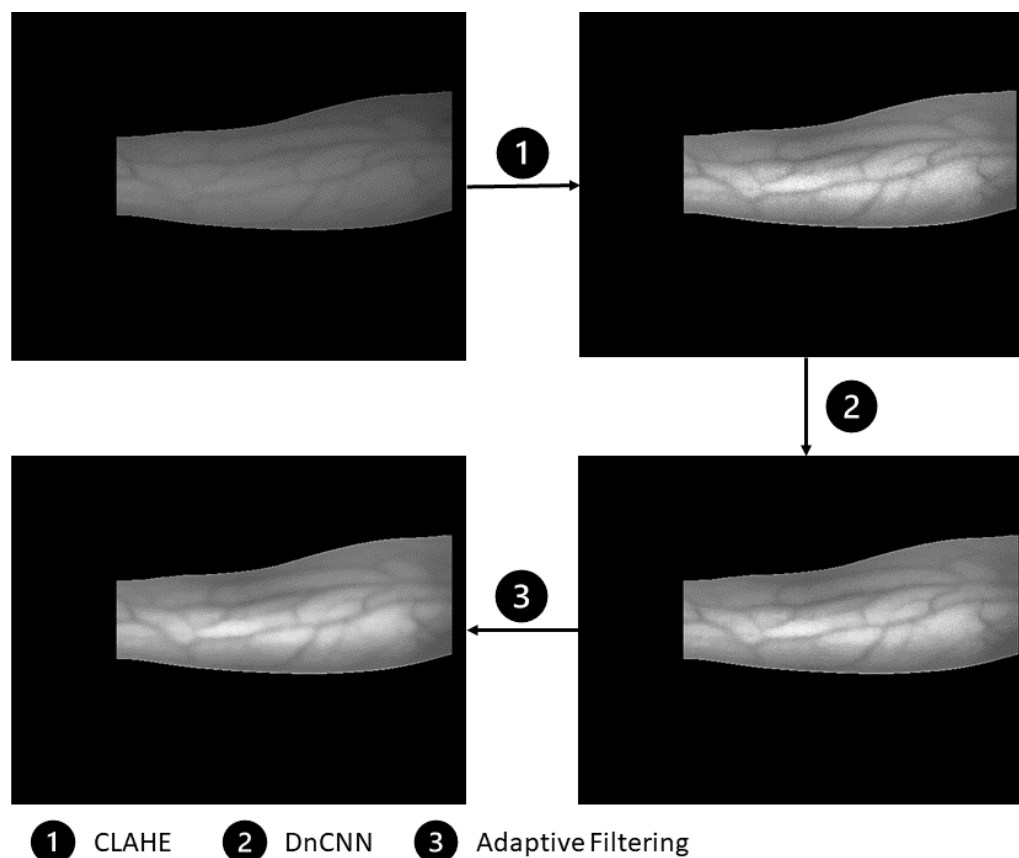


Figure 3.14: Visualizing Image Pre-processing.

3.5 Veins Feature Extraction Algorithms

3.5.1 State-of-the-art Algorithms

There were four types of state-of-the-art techniques implemented in this study, which are maximum curvature, repeated line tracking, principal curvature, and wide line detector. The working principles were discussed in Section 2.4 previously. The parameters selection of the respective algorithm were shown below.

Table 3.1: Parameters of Feature Extraction Algorithm.

Algorithm	Parameters
Repeated Line Tracking (Bram Ton, 2022a)	Maximum iterations = 2000; Distance between tracking point and profile = 6; Width of profile = 17
Maximum Curvature (Bram Ton, 2022a)	Sigma = 5
Principal Curvature (Christof Kauba, 2022)	Sigma = 4; The percentage used for hard thresholding = 4;
Wide Line Detector (Bram Ton, 2022b)	Radius of the circular neighbourhood region = 6 Neighbourhood threshold = 0.9 Sum of neighbourhood threshold = 40

The parameters were optimized through trial-and-error methods and the relationship between the parameters and the results. Since these algorithms were not given a guideline to choose the suitable parameters, the outcome of the veins segmentation results was observed and then tuned for optimization. The relationship between tuning the parameters was summarized below.

In repeated line tracking, the line tracking initiates at different positions. A cross-sectional profile is used to find the deepest point at the s-p-t valley. The width of the profile should be wider than the veins and not collide with other veins. The lines are connected between the tracking point and the profile, shorter profile would result in high processing time but is more accurate. The distance selected is nice enough to connect the lines without spending too much computational time. The number of iterations determines how likely the veins were detected. Iterations under the threshold would include the noise as veins while above that the thick veins would be detected more than thin veins.

The maximum curvature method calculates the local maximum in four different directions and the highest score of the curvature detected is identified as veins. The sigma parameter is to construct the filter kernels to determine the local minimum across the profile. Higher sigma would be insensitive to the small veins while noises would include if lower than that.

The principal curvature identifies the local minimum through the eigenvalues and eigenvectors of the Hessian Matrix. The sigma parameters determine the size of the Gaussian kernel to create and share the same function as maximum curvature. The hard threshold is used to remove the noises during gradient normalization. Hence, it is normally adjusted to a small value.

Lastly, the wide line detector uses a circular mask to match the pixels of the image. The veins are identified if the number of dark pixels in the circle exceeds the threshold which determines by the parameter, the sum of the neighbourhood threshold. The pixels are identified as ‘dark’ through the parameter of the neighbourhood threshold. The circular mask should be as close to the size of the veins so that the mask can cover all the veins in the image.

3.5.2 U-net Transfer Learning Model

3.5.2.1 Overview

The proposed method utilized the U-net model architecture, where the pre-trained model weights were transferred to the encoder part. A total of 28 pre-trained models from various families were used for training, including ResNet, ResNeXt, VGG, Inception, DenseNet, SENet, MobileNet, and EfficientNet. The pre-trained models were obtained from training on the ImageNet dataset and their weights were transferred to the proposed model.

3.5.2.2 Data Augmentation

To counterbalance the small dataset size, data augmentation was applied to the pre-processed datasets in order to expand the range and diversity of the training data. Augmentation strategies were chosen based on the expected variations during the injection process, and included:

- i) Horizontally flipping 50% of images,
- ii) Randomly cropping the width or height of some images by 10%.
- iii) Applying Gaussian blur with $\sigma = 0$ to 0.5 on 50% of images,
- iv) Changing the contrast of images by 75% to 150% of the original values,

- v) Adjusting the brightness of images by 80% to 120% of the original values, with 20% of images having different RGB channel values.
- vi) Rotating images by -10° to 10° .

3.5.2.3 Model Training Settings

The forearm vein dataset was divided by a ratio of 7:1:2 into three subsets, namely the training, validation, and testing sets. The resulting numbers of images in each set were 175 for training, 25 for validation, and 50 for testing.

To construct the U-Net model, a pre-trained model with weights was used as the encoder. The pre-trained model was obtained from a publicly available Github repository to facilitate the training process. (Pavel Iakubovskii, 2019). The last layer of the U-Net model was activated using the Sigmoid function to facilitate binary class prediction, as vein segmentation involves a binary classification task.

During the backpropagation process, the Adam optimizer was used and set at 0.0001 learning rate. Then, the model was trained for 50 epochs using a batch size of 8. The combination of binary cross-entropy loss and binary focal loss was used to evaluate the training losses in each epoch. The experiment was conducted on TensorFlow 2.0 framework and trained using Nvidia GTX1060.

3.6 Evaluation of Algorithm

Metrics evaluation is to identify whether the algorithm performed as expected. There were six types of metrics to determine the performance of the algorithm, which are accuracy, specificity, recall, precision, Dice coefficient, and Jaccard index.

These metrics were based on the fundamentals of the confusion matrix, which measures the predicted results and actual data. The feature extraction output generated by the algorithm is the predicted case while the ground truth data refers to the actual case.

		Actual Case	
		Positive	Negative
Predicted Case	Positive	TP	FP
	Negative	FN	TN

Figure 3.15: Confusion Matrix.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.1)$$

$$Specificity = \frac{TN}{TN + FP} \quad (3.2)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (3.3)$$

$$Precision = \frac{TP}{TP + FP} \quad (3.4)$$

$$Dice\ Coefficient = \frac{2TP}{2TP + FP + FN} \quad (3.5)$$

$$Jaccard\ Index = \frac{Dice\ Coefficient}{2 - Dice\ Coefficient} \quad (3.6)$$

A dice coefficient is designed to measure how much the predicted veins intersect with the actual case and penalize the score for falsely identified veins. The evaluation result compares the segmentation of veins and the ground truth where both are in binary representation. The Dice coefficient shares the same formula with the F1 score when evaluating binary results. It takes the balance between recall and precision and provides a harmonic mean of the performance.

3.7 Work Plan

This project was divided into two parts, each spanning 14 weeks. The first part focused on completing the progress report, and the schedule is outlined in Figure 3.16. The activities planned for this part included four major tasks. In the first four weeks, extensive literature research was conducted to understand the latest trends and state-of-the-art in vein segmentation algorithms, and

datasets were requested during this period. Over the next four weeks, an algorithm was developed based on the literature review. However, open-source code was available, so the development process was centered on learning how to use the algorithm. Additionally, ROI extraction of the forearm was carried out. Once the algorithm was developed, the datasets were used to check the segmentation results. The testing phase took longer than anticipated, with a duration of approximately five weeks instead of the planned two weeks. Finally, report writing was completed in week 11, which involved preparing the progress report and an oral presentation.

No.	Project Activities	Planned Completion Date	Weeks														
			W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	
1.	Literature Review	2022-07-08	█	█	█	█											
2.	Development of Algorithm	2022-08-05					█	█	█	█							
3.	Preliminary Testing	2022-08-19									█	█					
4.	Report Writing & Oral Presentation	2022-09-17											█	█	█	█	█

Figure 3.16: Gantt Chart of FYP Part 1.

Figure 3.17 shows the Gantt chart for part 2 of the project. The first four weeks were allocated for transfer learning model development, with each model requiring significant computational power and time. The timeline was set at four weeks as a safety line for algorithm development. The next four weeks were dedicated to finalizing the FYP report, with a focus on editing the content based on the proposed method. In week 11, a poster was required for submission to the FYP poster competition, which was completed unexpectedly quickly within a week. The last three weeks were spent enhancing the report and reviewing the developed algorithm.

No.	Project Activities	Planned Completion Date	Weeks														
			W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	
1.	Algorithm Development	2023-02-25	█	█	█	█											
2.	Thesis Writing	2023-03-19					█	█	█								
3.	Thesis Reviewed	2023-04-08								█	█	█					
4.	FYP Poster	2023-04-15										█	█				
5.	Thesis Enhancement and Final Checking	2023-05-06												█	█	█	

Figure 3.17: Gantt Chart of FYP Part 2.

3.8 Summary

This chapter discussed the relevant steps to develop a vein feature extraction algorithm. The first step was to request and obtain the dataset, which was then pre-processed to ensure its suitability for use in training and testing the algorithm. Next, state-of-the-art and proposed vein feature extraction algorithms were developed. Finally the performance of the algorithm were evaluated by comparing it to existing approaches, in order to determine the best-performing model. Overall, the project aimed to advance the field of hand vein feature extraction by developing a novel algorithm that could improve upon existing techniques.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

There are 4 state-of-the-art vein segmentation technique and proposed U-net transfer learning model developed and their results were presented below. The metrics are evaluated based on the segmented veins and ground truth. From the performance of the metrics, it can be evaluated whether these originated finger vein-based algorithms are suitable to be applied to forearm datasets. The proposed transfer learning model were tested on using different pre-trained model to evaluate the best performance model in segmenting the veins.

4.2 Results

4.2.1 State-of-the-art Algorithms

The results presented in Table 4.1 showed that all of the feature extraction algorithms evaluated achieved high levels of accuracy and specificity, with the maximum curvature algorithm performing the best overall, achieving accuracy and specificity scores of 98.03% and 99.41%, respectively. On the other hand, the principal curvature method had the lowest accuracy and specificity scores, with values of 91.76% and 92.02%, respectively. While the principal curvature method had a relatively high recall score of 76.71%, indicating that most true veins were correctly identified, its precision score was the lowest of all the algorithms, at only 14.99%. In contrast, the maximum curvature algorithm had the highest precision score (41.41%), but the lowest recall score, at 23.14%. However, it should be noted that none of the algorithms achieved a precision score greater than 50%.

Table 4.1: Accuracy, Specificity, Recall, and Precision of the State-of-the-art Vein Segmentation Algorithm. The best achieved results are shaded in green and worst results are shaded in orange.

Algorithm	Repeated Line	Maximum Curvature	Principal Curvature	Wide Line Detector

	Tracking			
Accuracy	96.55%	98.03%	91.76%	97.18%
Specificity	97.34%	99.41%	92.02%	98.38%
Recall	55.15%	23.14%	76.71%	31.27%
Precision	26.76%	41.41%	14.99%	25.44%

Table 4.2 presents the Dice coefficient and Jaccard index results for the four algorithms evaluated. While the repeated line tracking method did not achieve the highest recall or precision scores, it had the highest mean Dice coefficient and mean Jaccard index, at 35.14% and 21.58%, respectively, which is reasonable given that it had the second-highest recall and precision scores among the feature extraction algorithms. However, the repeated line tracking algorithm had the highest standard deviation, at 7.76%. In contrast, the principal curvature algorithm had the lowest mean Dice coefficient and mean Jaccard index, at only 24.66% and 14.25%, respectively. Importantly, the differences in mean and median scores among all the algorithms were within 0.82%, indicating that segmentation performance was relatively stable across the datasets.

Table 4.2: Dice Coefficient and Jaccard Index of the State-of-the-art Veins Extraction Algorithm. The best achieved results are shaded in green and worst results are shaded in orange.

Algorithm	Dice Coefficient			Jaccard Index		
	Mean	Median	Std Dev	Mean	Median	Std Dev
Repeated Line Tracking	35.14%	35.95%	7.76%	21.58%	21.91%	5.63%
Maximum Curvature	29.17%	29.71%	4.41%	17.16%	17.45%	2.98%
Principal Curvature	24.66%	24.24%	6.80%	14.25%	13.79%	4.71%
Wide Line Detector	27.50%	28.12%	7.56%	16.16%	16.36%	5.01%

4.2.2 U-net Transfer Learning Model

Table 4.3 presents the U-net model's performance with different pre-trained models as the backbone. All pre-trained networks achieved scores of above 98% in terms of accuracy and specificity. EfficientNetB2 had the highest accuracy score of 98.78%, while ResNet-18 had the lowest accuracy score of 98.31%. The difference between the highest and lowest accuracy scores was only 0.47%, indicating the high efficiency of pre-trained models. On the other hand, ResNet-18 had the highest specificity score of 99.96%, while ResNet-152 scored the lowest specificity of 99.54%.

Although the accuracy and specificity scores of all pre-trained models were similar, their precision and recall scores reveal a different story when it comes to selecting a better backbone for the U-net model. Among all models, SEResNet34 achieved the highest precision score of 83.78%, while ResNet-152 had the lowest precision score of 65.33%. In contrast, the best and worst performers in recall were different from precision. EfficientNetB3 achieved the highest recall score of 55.27%, while ResNet-18 had the lowest recall score of 9.83%. These findings suggest that besides considering accuracy and specificity, precision and recall scores can provide additional insights into the performance of the pre-trained models as backbones for the U-net model.

Table 4.3: Accuracy, Specificity, Recall, and Precision of the U-net Transfer Learning Model. The best achieved results are shaded in green and worst results are shaded in orange.

Backbone	Accuracy	Specificity	Precision	Recall
ResNeXt-50	98.65%	99.71%	73.04%	41.75%
ResNeXt-101	98.69%	99.78%	77.56%	40.09%
Inception-v3	98.69%	99.78%	77.56%	40.09%
Inception-ResNet-v2	98.71%	99.75%	76.06%	43.21%
DenseNet121	98.69%	99.78%	77.56%	40.09%
DenseNet169	98.67%	99.78%	76.71%	39.61%
DenseNet201	98.68%	99.69%	72.75%	44.73%
SEResNet18	98.63%	99.79%	76.21%	36.60%
SEResNet34	98.50%	99.92%	83.78%	22.74%

SEResNet50	98.67%	99.62%	70.07%	47.99%
SEResNet101	98.66%	99.64%	70.63%	46.15%
SEResNet152	98.68%	99.78%	77.30%	39.34%
SEResNext50	98.61%	99.55%	66.69%	47.96%
SEResNext101	98.69%	99.68%	72.45%	45.63%
MobileNet	98.77%	99.72%	76.13%	47.96%
MobileNetv2	98.59%	99.57%	66.56%	46.17%
EfficientNetB0	98.75%	99.60%	73.88%	48.76%
EfficientNetB1	98.75%	99.78%	78.51%	43.93%
EfficientNetB2	98.78%	99.74%	76.18%	45.43%
EfficientNetB3	98.75%	99.56%	70.25%	55.27%
EfficientNetB4	98.75%	99.65%	72.97%	50.61%
VGG16	98.77%	99.79%	79.87%	44.02%
VGG19	98.78%	99.75%	77.92%	46.44%
ResNet-18	98.31%	99.96%	83.60%	9.83%
ResNet-34	98.61%	99.74%	73.32%	38.21%
ResNet-50	98.59%	99.84%	75.46%	34.03%
ResNet-101	98.56%	99.84%	77.58%	29.68%
ResNet-152	98.57%	99.54%	65.33%	46.78%

The results of the segmentation performance using various pre-trained models as the backbone of U-net are presented in Table 4.4. It can be observed that the best and worst performers are consistent with the results obtained in the recall metrics. EfficientNetB3 achieved the highest dice coefficient and Jaccard index, with scores of 83.56% and 71.76%, respectively. On the other hand, ResNet-18 had the lowest dice coefficient of 70.11% and Jaccard index of 53.98%. These results suggest that the pre-trained models that perform well in recall metrics are likely to yield good results in Jaccard index and dice coefficient as well.

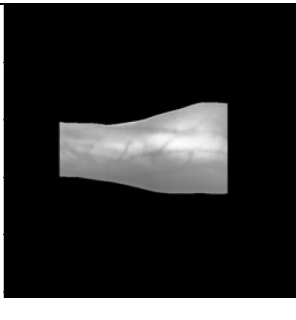
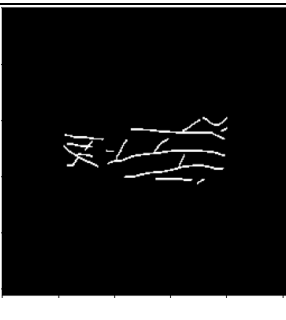
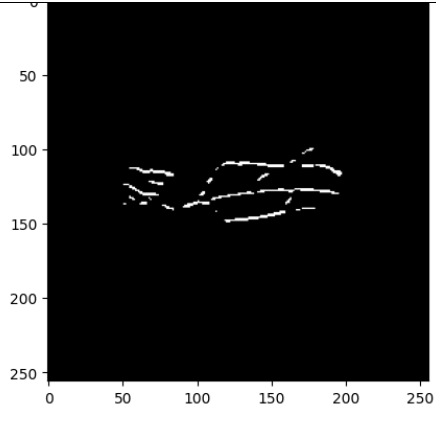
Table 4.4: Dice Coefficient and Jaccard Index of the U-net Transfer Learning Model. The best achieved results are shaded in green and worst results are shaded in orange.

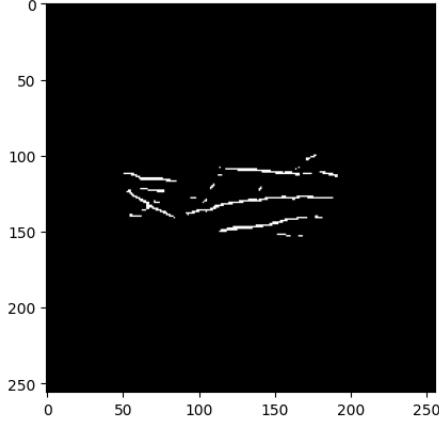
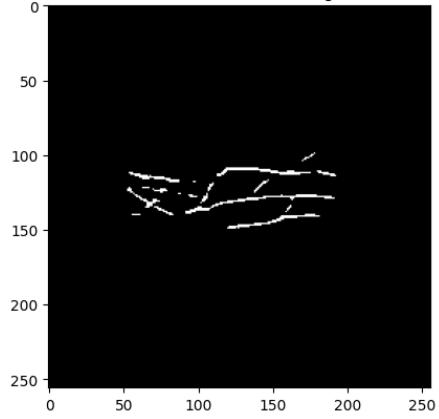
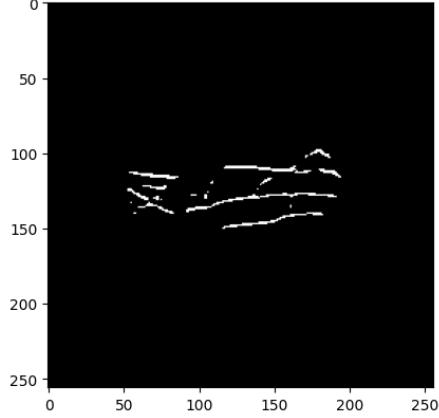
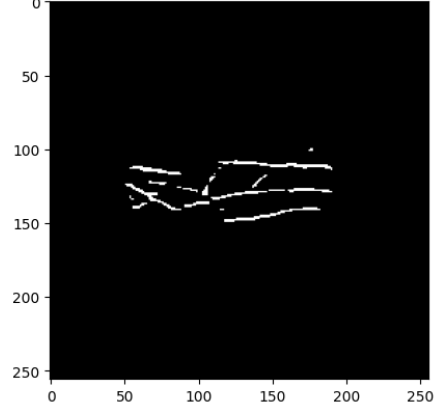
Backbone	Dice Coefficient	Jaccard Index
ResNeXt-50	80.53%	67.41%
ResNeXt-101	80.26%	67.02%
Inception-v3	80.46%	67.30%
Inception-ResNet-v2	81.21%	68.37%
DenseNet121	80.46%	67.30%
DenseNet169	80.25%	67.01%
DenseNet201	81.30%	68.49%
SEResNet18	79.33%	65.74%
SEResNet34	75.11%	60.14%
SEResNet50	81.83%	69.24%
SEResNet101	81.44%	68.68%
SEResNet152	80.22%	66.97%
SEResNext50	81.41%	68.65%
SEResNext101	81.50%	68.78%
MobileNet	82.51%	70.23%
MobileNetv2	80.97%	68.03%
EfficientNetB0	82.47%	70.16%
EfficientNetB1	81.64%	68.98%
EfficientNetB2	82.40%	70.07%
EfficientNetB3	83.56%	71.76%
EfficientNetB4	82.82%	70.68%
VGG16	81.79%	69.20%
VGG19	82.29%	69.90%
ResNet-18	70.11%	53.98%
ResNet-34	79.57%	66.08%
ResNet-50	78.50%	64.61%
ResNet-101	77.26%	62.94%
ResNet-152	80.96%	68.02%

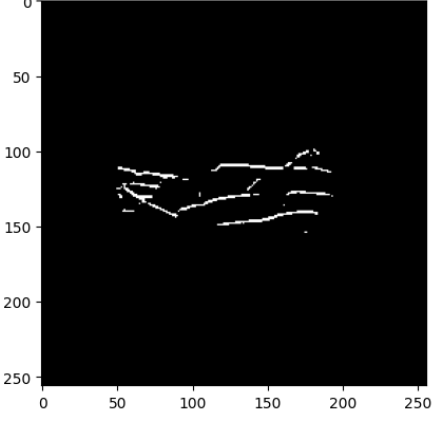
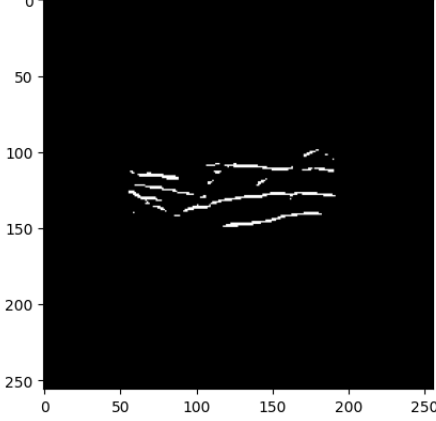
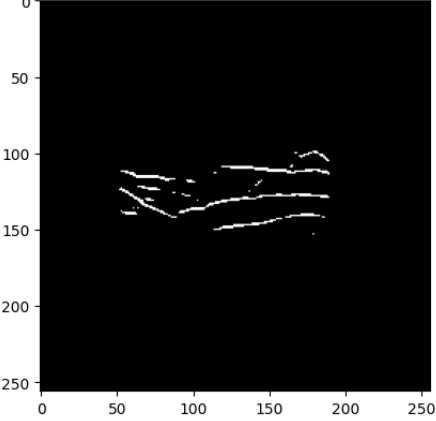
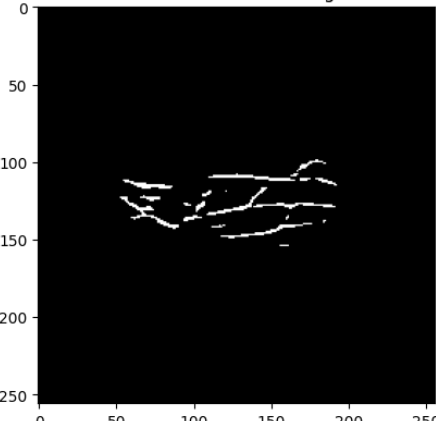
Table 4.5 presents the vein segmentation results obtained from each transfer learning model using different backbones. To identify the strengths and weaknesses of each backbone, the same pre-processed test image is used for comparison. According to the ground truth, there are three long horizontal veins visible in the image, which are the major veins for injection. All backbones perform well in segmenting these three vein lines.

However, there are two details that differentiate the performance of these backbones. Firstly, there is a problem with discontinuity in segmented lines for ResNeXt-50 and MobileNet, with ResNeXt-50 having more significant discontinuity where a vein could be cut into multiple short veins. Secondly, the detection of vertical veins is weaker for ResNeXt-50, Inception-v3, DenseNet201, MobileNet, EfficientNetB3, and VGG19. This could be due to the fact that vertical veins are less prominent than horizontal veins, leading to fewer features to train the model. However, the vertical veins are not as critical as the horizontal veins for injection, which makes this issue less severe.

Table 4.5: Samples of segmented veins generated by the top performer of each pre-trained model family.

<p>Test Image:</p> 	<p>Ground truth:</p> 
Backbone	Predicted Result
<p>ResNeXt-50</p>	

Inception-v3	 <p>A saliency map visualization for the Inception-v3 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
Inception-ResNet-v2	 <p>A saliency map visualization for the Inception-ResNet-v2 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
DenseNet201	 <p>A saliency map visualization for the DenseNet201 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
SEResNext101	 <p>A saliency map visualization for the SEResNext101 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>

MobileNet	 <p>A saliency map visualization for the MobileNet model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
EfficientNetB3	 <p>A saliency map visualization for the EfficientNetB3 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
VGG19	 <p>A saliency map visualization for the VGG19 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>
ResNet-152	 <p>A saliency map visualization for the ResNet-152 model. The image is a 256x256 grayscale plot with axes labeled from 0 to 250. The background is black, and the object is highlighted in white. The object is a horizontal, elongated shape with some internal structure, centered in the image.</p>

4.3 Discussion

4.3.1 Performance of State-of-the-art Hand Vein Segmentation

The use of accuracy and specificity metrics to evaluate segmentation algorithm performance is limited due to imbalanced datasets, where the background pixels make up more than half of the total pixels in an image. As a result, regions other than the region of interest contribute significantly to the accuracy and specificity scores. In contrast, the recall and precision metrics provide insight into how well the four algorithms perform in terms of identifying true veins. Recall is particularly relevant for evaluating feature extraction performance in vein segmentation, as it indicates how likely the algorithm is to identify all real veins (ground truth). However, in medical practice, precision is more crucial than recall, as false-positives can cause confusion for practitioners. Therefore, while recall is a suitable metric for evaluating feature extraction performance, precision is of greater importance in medical practice.

The results from Table 4.1 reveal two significant observations. First, the principal curvature method had the highest recall score, at 76.71%, but the lowest precision score, at 14.99%. This suggests that more than 80% of the segmented veins are not true veins, indicating poor performance in the vein segmentation task, as reflected in the lowest Dice coefficient and Jaccard index scores. Second, all of the state-of-the-art algorithms performed poorly on the NTUIFDB v1 Datasets, with all of their Dice coefficient and Jaccard index scores below 50%, as shown in Table 4.2. These state-of-the-art techniques were designed to segment finger veins datasets, and their performance may be affected when switching to forearm datasets, as the skin on the forearm is thicker, and the veins are less visible and have less contrast with the surrounding skin. Therefore, these state-of-the-art algorithms are not suitable for performing vein segmentation tasks on the NTUIFDB v1 Datasets.

4.3.2 Performance of U-net Transfer Learning Model

Table 4.6 illustrates the loss of the best performer of each pre-trained model family (same as Jaccard index) during training and validation stage. The selection of best performer is based on the Jaccard index from Table 4.4. For example, DenseNet family consists of DenseNet121, DenseNet169, and

DenseNet201 where DenseNet201 would be selected in Table 4.6 as it had the highest Jaccard index of 68.49%.

The training losses for all models were similar and slowly saturated at 40 epochs. In the first 10 epochs, the training loss curve experienced a steep decrease, after which the rate of decrease slowed down. However, there were differences in the performance of validation loss across the different backbones.

ResNeXt-50 experienced a steep decrease in validation loss in the first 20 epochs, which continued to drop until 50 epochs. The ultimate validation loss was higher than the training loss, indicating that the model was slightly overfit to the training data. However, the learning rate was good.

Inception-v3 exhibited a gradual decrease in validation loss over 20 epochs, eventually converging to the final validation loss. While the validation loss crossed over the training loss in the first 5 epochs, it continued to remain at a distance from the training loss thereafter. The model's learning rate was also considered good, with slight overfitting to the training data.

The Inception-ResNet-v2 model experienced a steep drop in validation loss in the first 20 epochs but started to intersect with the training loss afterwards, with a difference of less than 0.01. This indicates a good learning rate, but the model may be slightly overfitting to the training data.

As for DenseNet201, it also had a good learning rate, with a steep decrease in validation loss in the initial 20 epochs. The validation loss and training loss were separated by some distance, suggesting that the model was slightly overfitting.

MobileNet and SEResNext101 models had similar behavior, where the validation loss gradually decreased in the first 20 epochs. However, the validation loss of SEResNext101 remained closer to the training loss and intersected with it several times. The validation loss of MobileNet remained at a certain distance from the training loss throughout the 50 epochs.

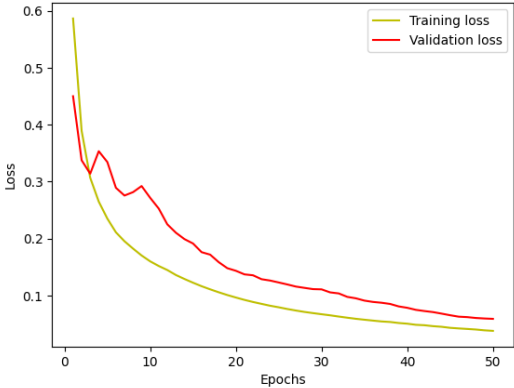
EfficientNetB3 had the best performance in Jaccard index, and its validation loss decreased linearly in the first 35 epochs, indicating a low learning rate. The validation loss then started to saturate, but did not intersect with the training loss, indicating that the model was still slightly overfitting to the training data.

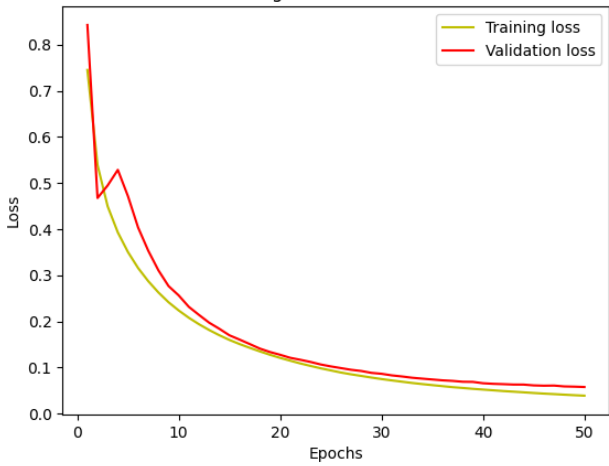
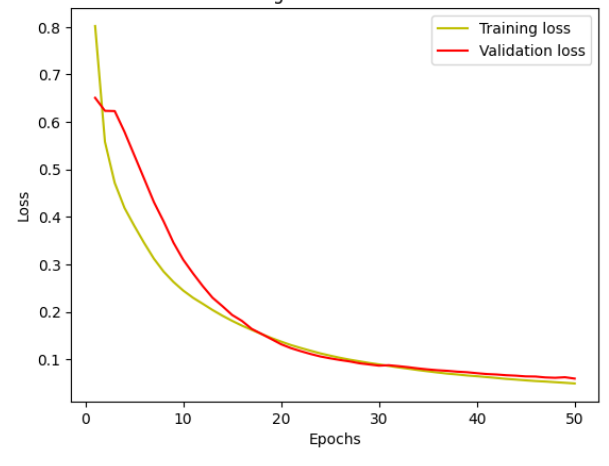
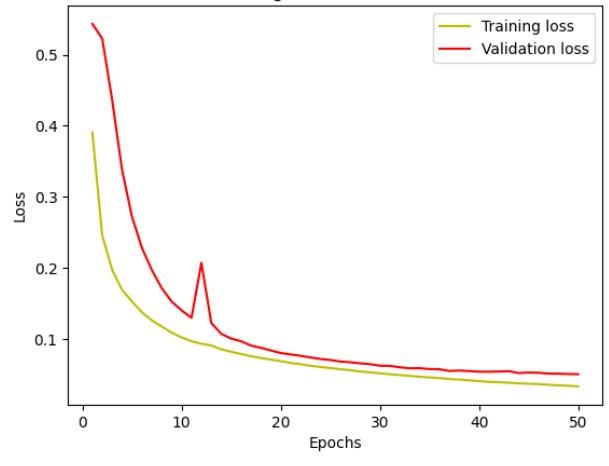
VGG19 had a V-shape valley in the first 5 epochs, with the validation loss crossing over the training loss. However, it quickly rose above the training loss and continued to saturate until the end. This behavior indicates that the model was trained at a good learning rate and exhibited a slightly overfitting behavior.

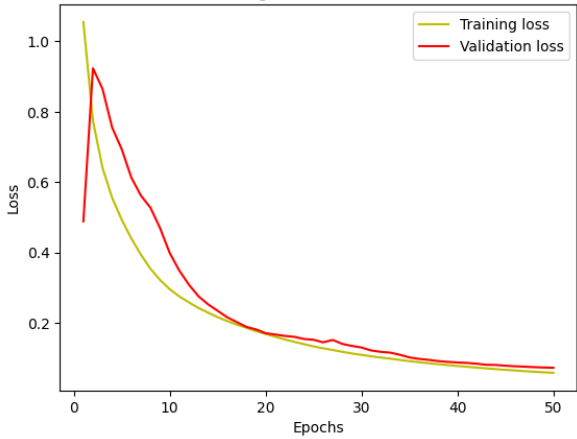
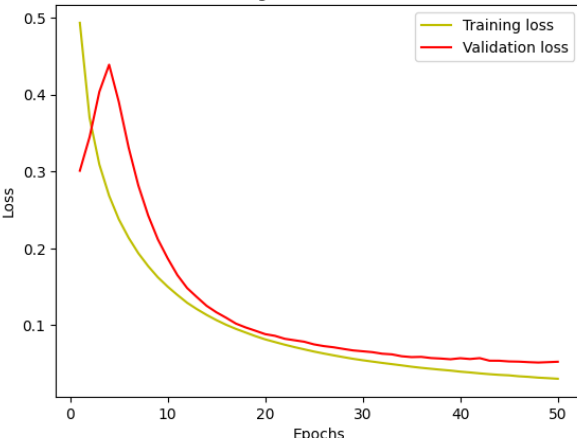
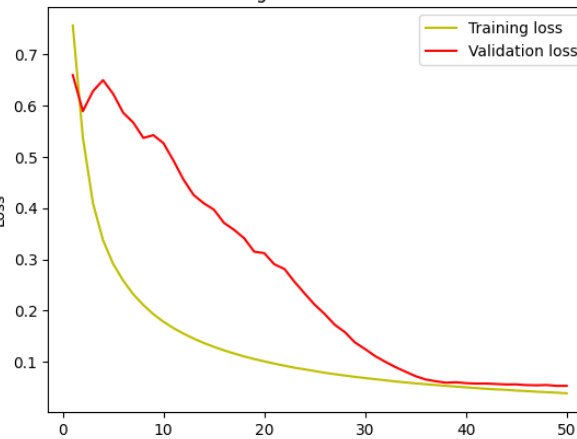
ResNet-152 had a steep decrease in validation loss in the first 10 epochs and then decreased slowly while moving along with the training loss with a certain distance. This behavior shows that the model had a good learning rate and was slightly overfitting to the training data.

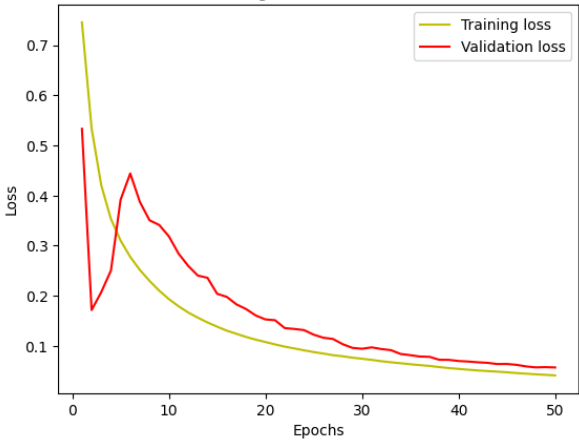
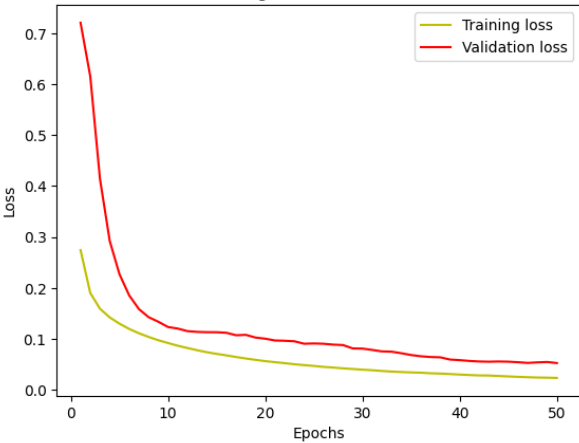
In short, the parameters selected to train the model was suitable to be used in training. As different model might have different optimized parameters, the learning rate and number of epochs used is considered a suitable selection that could observed the model performance under the same training conditions.

Table 4.6: Training and Validation Performance of the Top Performer Pre-trained Model in Each Pre-trained Model Family.

Backbone	Training & Validation Loss
ResNeXt-50	<p data-bbox="874 1133 1086 1155">Training and validation loss</p>  <p data-bbox="831 1570 1082 1599">Train loss = 0.0377</p> <p data-bbox="842 1626 1070 1655">Val loss = 0.0588</p>

Inception-v3	<p>Training and validation loss</p>  <p>Train loss = 0.0389 Val loss = 0.0577</p>
Inception-ResNet-v2	<p>Training and validation loss</p>  <p>Train loss = 0.0495 Val loss = 0.0597</p>
DenseNet201	<p>Training and validation loss</p>  <p>Train loss = 0.0330 Val loss = 0.0501</p>

SEResNext101	<p>Training and validation loss</p>  <p>Train loss = 0.0582 Val loss = 0.0726</p>
MobileNet	<p>Training and validation loss</p>  <p>Train loss = 0.0303 Val loss = 0.0524</p>
EfficientNetB3	<p>Training and validation loss</p>  <p>Train loss = 0.0383 Val loss = 0.0529</p>

VGG19	<p style="text-align: center;">Training and validation loss</p>  <p style="text-align: center;">Train loss = 0.0411 Val loss = 0.0569</p>
ResNet-152	<p style="text-align: center;">Training and validation loss</p>  <p style="text-align: center;">Train loss = 0.0235 Val loss = 0.0527</p>

After analyzing the performance of different backbone models during transfer learning in Table 4.4, a clear trend has emerged. EfficientNet family outperforms other families in terms of Jaccard index, with all but one EfficientNet family achieving more than 70%. In contrast, ResNet family did not perform well as the backbone during transfer learning, with ResNet18 having the lowest Jaccard index of 53.98%. Three out of five ResNets scored lower than 65% of Jaccard index. Since ResNets were designed to solve the vanishing gradient problem by introducing shortcuts to the layers, they may not be powerful enough to perform vein segmentation tasks (He et al., 2015). In contrast, EfficientNet uses compound scaling to efficiently scale the dimensions of depth, width, and resolution using computing resources (Tan

and Le, 2019b). Therefore, EfficientNet has more capability to learn complex features from the input image, which makes it outperform other backbone models. This trend suggests that choosing the right backbone architecture is critical to achieving high performance in vein segmentation tasks.

4.3.3 Performance Analysis of State-of-the-art Algorithms and the Proposed Models

Table 4.7 has presented the proposed method as compared with the state-of-the-art algorithm. The datasets used and pre-processing techniques applied were the same before applying the vein segmentation algorithm. Therefore, the comparison can reflect on the actual performance of each method without considering other variable factor.

Table 4.7: Performance Comparison between State-of-the-art Algorithm and the Proposed Methods, where only the best backbone model is selected.

Method	Precision	Recall	Dice Coefficient	Jaccard Index
Repeated Line Tracking	26.76%	56.15%	35.14%	21.58%
Maximum Curvature	41.41%	23.14%	29.17%	17.16%
Principal Curvature	14.99%	76.71%	24.66%	14.25%
Wide Line Detector	25.44%	31.27%	27.50%	16.16%
U-net Transfer Learning (EfficientNetB3 as backbone)	70.25%	55.27%	83.56%	71.76%

The proposed method with various backbone models, as shown in Table 4.7, outperformed the state-of-the-art techniques. However, the

comparison would only select the best performer backbone. The proposed method achieves a precision 70.25%, which is significantly higher than the state-of-the-art maximum curvature method with a precision of 41.41%. It means that the proposed method is 30% more reliable in predicting pixel belonging to veins. Although the proposed method's recall score of 55.27% is lower than repeated line tracking and principal curvature methods, those state-of-the-art methods have low precision, indicating that most segmented veins are not actual veins. Dice coefficient and Jaccard index reflect the overall performance of the segmentation results. From the experiment, the proposed method achieves an 83.56% dice coefficient and a 71.76% Jaccard index, which is much more reliable than the best of state-of-the-art methods, repeated line tracking, with a dice coefficient of 35.14% and a Jaccard index of 21.58%.

The comparison results established the superiority and robustness of the proposed method in vein segmentation tasks. Deep learning models are found to be more suitable for this purpose compared to handcrafted feature extraction techniques, which were developed based on specific datasets and may not perform well on different datasets. Moreover, the state-of-the-art algorithms were designed to segment finger veins and their performance may not be robust when applied to forearm veins. In contrast, the proposed method can be further fine-tuned through transfer learning to learn features on new datasets, making it theoretically possible to achieve accurate vein segmentation on different datasets.

4.4 Summary

The performance of four state-of-the-art algorithms was found to be poor when evaluated on the NTUIFDB v1 forearm vein dataset. On the other hand, the proposed transfer learning model which used U-net as the architecture and EfficientNetB3 as the backbone demonstrated excellent performance in vein segmentation with scores of 83.56% for Dice coefficient and 71.76% for Jaccard index. These results suggest that the proposed model has the potential to be used in clinical settings for accurate and reliable vein segmentation.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

5.1 Conclusions

In conclusion, this study has demonstrated the effectiveness of transfer learning in addressing the small dataset problem while ensuring high-performance levels in forearm vein feature extraction. Our results indicate that the EfficientNetB3 model, used as the backbone of the U-net architecture, outperformed other state-of-the-art techniques, achieving the highest accuracy rates in segmenting the NTUIFDB v1 dataset. Additionally, our findings highlight the limitations of handcrafted vein feature extraction methods, which did not perform as well as deep learning models in extracting relevant features of forearm veins. These results underscore the potential of deep learning techniques for improving venous access success rates and reducing healthcare costs associated with difficult venous access. In summary, this study provides valuable insights into the application of transfer learning and deep learning for vein feature extraction in the forearm, paving the way for further research in this important area of medical imaging.

5.2 Recommendations for future work

There are several potential areas for future research in the field of forearm vein feature extraction. Firstly, it is recommended that the feature learning capability of the transfer learning model be further improved by using larger datasets. The use of the NTUIFDB v2 dataset, which includes a larger number of images, could significantly enhance the model's performance. However, this dataset requires healthcare professionals to label the ground truth of the images, which can be time-consuming and labor-intensive.

Secondly, it is recommended that future studies should focus on training the model on open-set scenarios. This would enable the development of a robust vein feature extraction algorithm that could identify veins from different parts of the body, not just the forearm. Such a model could be especially useful in emergency medical situations, where venous access is

often challenging due to factors such as patient movement, time constraints, and poor lighting conditions.

Overall, further research in these areas has the potential to significantly advance the field of medical imaging and improve the accuracy and reliability of vein-based feature extraction systems.

REFERENCES

- Anon. 2022. *Nitroglycerin Sublingual: MedlinePlus Drug Information*. [online] Available at: <<https://medlineplus.gov/druginfo/meds/a601086.html>> [Accessed 25 August 2022].
- Armenteros-Yeguas, V., Gárate-Echenique, L., Tomás-López, M.A., Cristóbal-Domínguez, E., Moreno-de Gusmão, B., Miranda-Serrano, E. and Moraza-Dulanto, M.I., 2017. Prevalence of difficult venous access and associated risk factors in highly complex hospitalised patients. *Journal of Clinical Nursing*, [online] 26(23–24), p.4267. <https://doi.org/10.1111/JOCN.13750>.
- Atalay, H., Erbay, H., Tomatir, E., Serin, S. and Oner, O., 2005. The use of transillumination for peripheral venous access in paediatric anaesthesia [4]. *European Journal of Anaesthesiology*, [online] 22(4), pp.317–318. <https://doi.org/10.1017/S026502150524053X>.
- Azuetto-Ríos, Á., Hernández-Gómez, L.-E. and Hernández-Santiago, K.-A., 2016. Forearm and Hand Vein Detection System for an Infrared Image Database. *Research in Computing Science*, 127, pp.137–147.
- Bains, K.N.S. and Lappin, S.L., 2021. Anatomy, Shoulder and Upper Limb, Elbow Cubital Fossa. *StatPearls*. [online] Available at: <<https://www.ncbi.nlm.nih.gov/books/NBK459250/>> [Accessed 3 September 2022].
- Balter, M.L., Chen, A.I., Maguire, T.J. and Yarmush, M.L., 2015. The System Design and Evaluation of a 7-DOF Image-Guided Venipuncture Robot. *IEEE transactions on robotics : a publication of the IEEE Robotics and Automation Society*, [online] 31(4), p.1044. <https://doi.org/10.1109/TRO.2015.2452776>.
- Bram Ton, 2022a. *Miura et al. vein extraction methods*. [online] MATLAB Central File Exchange. Available at: <https://www.mathworks.com/matlabcentral/fileexchange/35716-miura-et-al-vein-extraction-methods?s_tid=ta_fx_results> [Accessed 10 September 2022].

Bram Ton, 2022b. *Wide line detector. MATLAB Central File Exchange.* Available at: <<https://www.mathworks.com/matlabcentral/fileexchange/35754-wide-line-detector>> [Accessed 10 September 2022].

Chayovan, T., Limumpornpetch, P. and Hongsakul, K., 2019. Success rate of adrenal venous sampling and predictors for success: a retrospective study. *Polish Journal of Radiology*, [online] 84, p.e136. <https://doi.org/10.5114/PJR.2019.84178>.

Cheng, Z., Davies, B.L., Caldwell, D.G. and Mattos, L.S., 2016. A venipuncture detection system for robot-Assisted intravenous catheterization. *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2016-July, pp.80–86. <https://doi.org/10.1109/BIROB.2016.7523602>.

Choi, J.H., Song, W., Kim, T., Lee, S.-R. and Kim, H.C., 2009. Finger vein extraction using gradient normalization and principal curvature. *Image Processing: Machine Vision Applications II*, 7251, p.725111. <https://doi.org/10.1117/12.810458>.

Choras, R.S., 2017. Personal identification using forearm vein patterns. *2017 International Work Conference on Bio-Inspired Intelligence: Intelligent Systems for Biodiversity Conservation, IWOBI 2017 - Proceedings.* <https://doi.org/10.1109/IWOBI.2017.7985519>.

Christof Kauba, 2022. *PLUS OpenVein Finger- and Hand-Vein Toolkit.* [online] The Multimedia Signal Processing and Security Lab. Available at: <<https://www.wavelab.at/sources/OpenVein-Toolkit/>> [Accessed 10 September 2022].

Crisan, S., Tarnovan, I.G. and Crişan, T.E., 2007. A low cost vein detection system using near infrared radiation. *Proceedings of the 2007 IEEE Sensors Applications Symposium, SAS.* <https://doi.org/10.1109/SAS.2007.374359>.

Cuper, N.J., Verdaasdonk, R.M., de Roode, R., de Vooght, K.M.K., Viergever, M.A., Kalkman, C.J. and de Graaff, J.C., 2011. Visualizing veins with near-infrared light to facilitate blood withdrawal in children. *Clinical Pediatrics*, [online] 50(6), pp.508–512. <https://doi.org/10.1177/0009922810395932>.

Drake, R.L., Vogl, A.W. and Mitchell, A.W.M., 2018. *Gray's Basic Anatomy, Second Edition*. 2nd ed. Elsevier, Inc. Elsevier, Inc.

Fairuz, S., Habaebi, M.H. and Elsheikh, E.M.A., 2018. Finger Vein Identification Based on Transfer Learning of AlexNet. *Proceedings of the 2018 7th International Conference on Computer and Communication Engineering, ICCCE 2018*, pp.465–469. <https://doi.org/10.1109/ICCCE.2018.8539256>.

Gallieni, M., Pittiruti, M. and Biffi, R., 2008. Vascular access in oncology patients. *CA: a cancer journal for clinicians*, [online] 58(6), pp.323–346. <https://doi.org/10.3322/CA.2008.0015>.

Garcia-Martin, R. and Sanchez-Reillo, R., 2021. Deep learning for vein biometric recognition on a smartphone. *IEEE Access*, [online] 9, pp.98812–98832. <https://doi.org/10.1109/ACCESS.2021.3095666>.

Girgis, K.K., 2014. Ultrasound guidance versus transillumination for peripheral intravenous cannulation in pediatric patients with difficult venous access. *The Egyptian Journal of Cardiothoracic Anesthesia*, [online] 8(1), p.39. <https://doi.org/10.4103/1687-9090.137236>.

Hatfield, A. and Bodenham, A., 1999. Portable ultrasound for difficult central venous access. *British Journal of Anaesthesia*, 82(6), pp.822–826. <https://doi.org/10.1093/BJA/82.6.822>.

He, K., Zhang, X., Ren, S. and Sun, J., 2015. Deep Residual Learning for Image Recognition. [online] Available at: <http://image-net.org/challenges/LSVRC/2015/> [Accessed 29 April 2023].

Horowitz, S.H., 2000. Venipuncture-induced causalgia: anatomic relations of upper extremity superficial veins and nerves, and clinical considerations.

Transfusion, [online] 40(9), pp.1036–1040. <https://doi.org/10.1046/J.1537-2995.2000.40091036.X>.

Hu, J., Shen, L., Albanie, S., Sun, G. and Wu, E., 2017. Squeeze-and-Excitation Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, [online] 42(8), pp.2011–2023. <https://doi.org/10.1109/TPAMI.2019.2913372>.

Huang, B., Dai, Y., Li, R., Tang, D. and Li, W., 2010. Finger-vein authentication based on wide line detector and pattern normalization. *Proceedings - International Conference on Pattern Recognition*, pp.1269–1272. <https://doi.org/10.1109/ICPR.2010.316>.

Huang, G., Liu, Z., Van Der Maaten, L. and Weinberger, K.Q., 2016. Densely Connected Convolutional Networks. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, [online] 2017-January, pp.2261–2269. <https://doi.org/10.1109/CVPR.2017.243>.

Huynh, N.Q., Xu, X., Kong, A.W.K. and Subbiah, S., 2015. A preliminary report on a full-body imaging system for effectively collecting and processing biometric traits of prisoners. *IEEE Workshop on Computational Intelligence in Biometrics and Identity Management, CIBIM, 2015-January(January)*, pp.167–174. <https://doi.org/10.1109/CIBIM.2014.7015459>.

Ichimura, M., Sasaki, S. and Ogino, T., 2020. Tapping enhances vasodilation for venipuncture even in individuals with veins that are relatively difficult to palpate. *Clinical anatomy (New York, N.Y.)*, [online] 33(3), pp.440–445. <https://doi.org/10.1002/CA.23559>.

Jian, K.H.X.Z.S.R., 2015. Deep Residual Learning for Image Recognition. *Enzyme and Microbial Technology*, [online] 19(2), pp.107–117. Available at: <<http://image-net.org/challenges/LSVRC/2015/>> [Accessed 30 April 2023].

Jing, C.X., Meng, G.C., Tyng, C.M., Aluwee, S.A.Z.B.S. and Voon, W.P., 2021. An Automatic Vein Detection System Using Deep Learning for Intravenous (IV) Access Procedure. *1st National Biomedical Engineering*

Conference, NBEC 2021, pp.106–111.
<https://doi.org/10.1109/NBEC53282.2021.9618752>.

Kauba, C. and Uhl, A., 2018. Shedding light on the veins-reflected light or transillumination in hand-vein recognition. *Proceedings - 2018 International Conference on Biometrics, ICB 2018*, pp.283–290.
<https://doi.org/10.1109/ICB2018.2018.00050>.

Kim, H.J., Park, S.K. and Park, S.H., 2017. Upper limb nerve injuries caused by intramuscular injection or routine venipuncture. *Anesthesia and Pain Medicine*, 12(2), pp.103–110. <https://doi.org/10.17085/APM.2017.12.2.103>.

Knutsson, H., Westin, C.-F. and Kikinis, R., 2000. Adaptive image filtering. *Handbook of medical imaging*, pp.19–31.

Kuhns, L.R., Martin, A.J., Gildersleeve, S. and Poznanski, A.K., 1975. Intense Transillumination for Infant Venipuncture1. <https://doi.org/10.1148/116.3.734>, [online] 116(3), pp.734–735. <https://doi.org/10.1148/116.3.734>.

Kumar, A. and Zhou, Y., 2012. Human identification using finger images. *IEEE Transactions on Image Processing*, 21(4), pp.2228–2244.
<https://doi.org/10.1109/TIP.2011.2171697>.

Kuzu, R.S., Maiorana, E. and Campisi, P., 2020. Vein-based Biometric Verification using Transfer Learning. *2020 43rd International Conference on Telecommunications and Signal Processing, TSP 2020*, pp.403–409.
<https://doi.org/10.1109/TSP49548.2020.9163491>.

Lee, H., Lee, S.H., Kim, S.J., Choi, W.I., Lee, J.H. and Choi, I.J., 2015. Variations of the cubital superficial vein investigated by using the intravenous illuminator. *Anatomy & Cell Biology*, [online] 48(1), p.62.
<https://doi.org/10.5115/ACB.2015.48.1.62>.

Lee, T.C., Kashyap, R.L. and Chu, C.N., 1994. Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6), pp.462–478.
<https://doi.org/10.1006/CGIP.1994.1042>.

Leli, V.M., Rubashevskii, A., Sarachakov, A., Rogov, O. and Dylov, D. v., 2020. Near-Infrared-to-Visible Vein Imaging via Convolutional Neural Networks and Reinforcement Learning. *16th IEEE International Conference on Control, Automation, Robotics and Vision, ICARCV 2020*, pp.434–441. <https://doi.org/10.1109/ICARCV50220.2020.9305503>.

Liu, L., Zhang, D. and You, J., 2007. Detecting wide lines using isotropic nonlinear filtering. *IEEE Transactions on Image Processing*, 16(6), pp.1584–1595. <https://doi.org/10.1109/TIP.2007.894288>.

van Loon, F.H.J., Buise, M.P., Claassen, J.J.F., Dierick-van Daele, A.T.M. and Bouwman, A.R.A., 2018. Comparison of ultrasound guidance with palpation and direct visualisation for peripheral vein cannulation in adult patients: a systematic review and meta-analysis. *British Journal of Anaesthesia*, 121(2), pp.358–366. <https://doi.org/10.1016/J.BJA.2018.04.047>.

Marattukalam, F. and Abdulla, W.H., 2020. Segmentation of Palm Vein Images Using U-Net; Segmentation of Palm Vein Images Using U-Net. *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*.

Marie Gardiner, 2020. *Noise in Digital Photos: Where It Comes From and How to Fix It*. [online] Available at: <<https://photography.tutsplus.com/articles/what-is-digital-image-noise--cms-24554>> [Accessed 9 September 2022].

Maruyama, K., Mizuuchi, N., Moroi, T., Imura, C., Isobe, I., Usagawa, T. and Iriguchi, N., 2007. Non-invasive 3.0T MR angiography of peripheral arteries and veins using 3D-HASTE. *2007 IEEE/ICME International Conference on Complex Medical Engineering, CME 2007*, pp.570–573. <https://doi.org/10.1109/ICCME.2007.4381800>.

Mbamalu, D. and Banerjee, A., n.d. Classic techniques in medicine Methods of obtaining peripheral venous access in difficult situations.

Mcintosh, M.A., Shahani, U., Boulton, R.G. and McCulloch, D.L., 2010. Absolute Quantification of Oxygenated Hemoglobin within the Visual Cortex with Functional Near Infrared Spectroscopy (fNIRS). *Investigative Ophthalmology & Visual Science*, 51(9), pp.4856–4860. <https://doi.org/10.1167/IOVS.09-4940>.

Miura, N., Nagasaka, A. and Miyatake, T., 2004. Digital Object Identifier (Feature extraction of finger-vein patterns based on repeated line tracking and its application to personal identification. *Machine Vision and Applications*, pp.194–203. <https://doi.org/10.1007/s00138-004-0149-2>.

Miura, N., Nagasaka, A. and Miyatake, T., 2007. Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE Transactions on Information and Systems*, E90-D(8), pp.1185–1194. <https://doi.org/10.1093/IETISY/E90-D.8.1185>.

Mohd Hakim Bin Mohamed Ashri, 2021. Perceived confidence in performing peripheral venipuncture among dental practitioners in New Zealand and Malaysia (Thesis, Doctor of Clinical Dentistry). *University of Otago*. [online] Available at: <<http://hdl.handle.net/10523/12526>> [Accessed 22 August 2022].

Moore, K.L. and Rohen, W.J., 2013. *Clinically Oriented Anatomy*. [online] Lippincott Williams & Wilkins. Available at: <<https://books.google.com.my/books?id=Y0PfgEACAAJ>>.

Nee, P.A., Pictou, A.J., Ralston, D.R. and Perks, A.G.B., 1994. Facilitation of peripheral intravenous access: an evaluation of two methods to augment venous filling. *Annals of Emergency Medicine*, [online] 24(5), pp.944–946. [https://doi.org/10.1016/S0196-0644\(94\)70211-X](https://doi.org/10.1016/S0196-0644(94)70211-X).

Nguyen, J.D. and Duong, H., 2021. Anatomy, Shoulder and Upper Limb, Veins. *StatPearls*. [online] Available at: <<https://www.ncbi.nlm.nih.gov/books/NBK546676/>> [Accessed 3 September 2022].

NHS, 2022. *MRI scan - Who can have one - NHS*. [online] National Health Service. Available at: <<https://www.nhs.uk/conditions/mri-scan/who-can-have-it/>> [Accessed 29 August 2022].

Pan, C.T., Francisco, M.D., Yen, C.K., Wang, S.Y. and Shiue, Y.L., 2019. Vein Pattern Locating Technology for Cannulation: A Review of the Low-Cost Vein Finder Prototypes Utilizing near Infrared (NIR) Light to Improve Peripheral Subcutaneous Vein Selection for Phlebotomy. *undefined*, [online] 19(16), p.3573. <https://doi.org/10.3390/S19163573>.

Pavel Iakubovskii, 2019. *Segmentation Models*. Available at: <https://github.com/qubvel/segmentation_models> [Accessed 30 April 2023].

Qin, H., el Yacoubi, M.A., Lin, J. and Liu, B., 2019. An Iterative Deep Neural Network for Hand-Vein Verification. *IEEE Access*, 7, pp.34823–34837. <https://doi.org/10.1109/ACCESS.2019.2901335>.

Roberge, R.J., Kelly, M., Evans, T.C., Hobbs, E., Sayre, M. and Cottington, E., 1987. Facilitated intravenous access through local application of nitroglycerin ointment. *Annals of Emergency Medicine*, 16(5), pp.546–549. [https://doi.org/10.1016/S0196-0644\(87\)80682-0](https://doi.org/10.1016/S0196-0644(87)80682-0).

Ronneberger, O., Fischer, P. and Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, [online] 9351, pp.234–241. Available at: <<http://lmb.informatik.uni-freiburg.de/>> [Accessed 6 September 2022].

Sahana D S, Dayanand Lal. N, Vidya J and Bhanujyothi H C, 2020. An Efficient Scheme for Vein Detection using Accuvein Apparatus Based on Near Infrared with Broadcom Chip. *International Journal of Engineering and Advanced Technology*, [online] 9(3), pp.2181–2186. <https://doi.org/10.35940/IJEAT.C5616.029320>.

Sakudo, A., 2016. Near-infrared spectroscopy for medical applications: Current status and future perspectives. *Clinica Chimica Acta*, 455, pp.181–188. <https://doi.org/10.1016/J.CCA.2016.02.009>.

Scott, W.L., 1995. Central Venous Catheters: An Overview of Food and Drug Administration Activities. *Surgical Oncology Clinics of North America*, 4(3), pp.377–393. [https://doi.org/10.1016/S1055-3207\(18\)30433-2](https://doi.org/10.1016/S1055-3207(18)30433-2).

Shah, Z., Shah, S.A.A., Shahzad, A., Fayyaz, A., Khaliq, S., Zahir, A. and Meng, G.C., 2022. Deep Learning-Based Forearm Subcutaneous Veins Segmentation. *IEEE Access*, 10, pp.42814–42820. <https://doi.org/10.1109/ACCESS.2022.3167691>.

Sheng, M.Y., Zhao, Y., Liu, F.Q., Hu, Q.D., Zhang, D.W. and Zhuang, S.L., 2011. Acquisition and preprocessing of hand vein image. *Proceedings - 2011 4th International Symposium on Knowledge Acquisition and Modeling, KAM 2011*, pp.251–253. <https://doi.org/10.1109/KAM.2011.74>.

Simons, P., Smith, P.C., Lees, W.R. and Mcgrouter, D.A., 1996. Venous pumps of the hand: Their clinical importance. *The Journal of Hand Surgery: British & European Volume*, 21(5), pp.595–599. [https://doi.org/10.1016/S0266-7681\(96\)80137-9](https://doi.org/10.1016/S0266-7681(96)80137-9).

Simonyan, K. and Zisserman, A., 2015. VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION. *ICLR 2015*. [online] Available at: <<http://www.robots.ox.ac.uk/>> [Accessed 1 May 2023].

Szegedy, C., Liu, W., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., 2014. Going deeper with convolutions.

Tan, M. and Le, Q. V., 2019a. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *36th International Conference on Machine Learning, ICML 2019*, [online] 2019-June, pp.10691–10700. Available at: <<https://arxiv.org/abs/1905.11946v5>> [Accessed 1 May 2023].

Tan, M. and Le, Q. V., 2019b. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *36th International Conference on Machine*

Learning, ICML 2019, [online] 2019-June, pp.10691–10700. Available at: <<https://arxiv.org/abs/1905.11946v5>> [Accessed 29 April 2023].

Thomas P. Duffy, Gerald T. Harder and Kevin Mori, 2012. *Medical imaging device*. [online] US20120101343A1. Available at: <<https://appft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&p=1&u=/netahtml/PTO/srchnum.html&r=1&f=G&l=50&d=PG01&s1=20120101343.PGNR.>> [Accessed 29 August 2022].

Török, T., Bari, F., Kardos, A., Paprika, D. and Rudas, L., 1997. Isometric handgrip exercise-induced muscarinic vasodilation in the human skin microvasculature. *Acta Physiologica Hungarica*, [online] 85(3), pp.193–198. Available at: <<https://europepmc.org/article/med/10101534>> [Accessed 25 August 2022].

Torrey, L. and Shavlik, J., 2010. Transfer Learning. <https://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-60566-766-9.ch011>, [online] pp.242–264. <https://doi.org/10.4018/978-1-60566-766-9.CH011>.

Veinlite, 2022. *Clinically Proven Vein Finders for IV Access*. [online] Veinlite. Available at: <<https://www.veinlite.com/>> [Accessed 30 August 2022].

Waite, C., Waite, P. and Pirmohamed, M., 2004. Intravenous therapy. *Postgraduate Medical Journal*, [online] 80(939), pp.1–6. <https://doi.org/10.1136/PMJ.2003.010421>.

Walsh, G., 2008. Difficult Peripheral Venous Access: Recognizing and Managing the Patient at Risk. *Journal of the Association for Vascular Access*, 13(4), pp.198–203. <https://doi.org/10.2309/JAVA.13-4-7>.

Wu, W., Elliott, S.J., Lin, S. and Yuan, W., 2019. Low-cost biometric recognition system based on NIR palm vein image. *IET Biometrics*, [online] 8(3), pp.206–214. <https://doi.org/10.1049/IET-BMT.2018.5027>.

Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K. and San Diego, U., 2017. Aggregated Residual Transformations for Deep Neural Networks. [online]

Available at: <<https://github.com/facebookresearch/ResNeXt>> [Accessed 1 May 2023].

Yamada, K., Yamada, K., Katsuda, I. and Hida, T., 2008. Cubital fossa venipuncture sites based on anatomical variations and relationships of cutaneous veins and nerves. *Clinical anatomy (New York, N.Y.)*, [online] 21(4), pp.307–313. <https://doi.org/10.1002/CA.20622>.

Zhang, H., Tang, C., Kong, A.W.K. and Craft, N., 2012. Matching vein patterns from color images for forensic investigation. *2012 IEEE 5th International Conference on Biometrics: Theory, Applications and Systems, BTAS 2012*, pp.77–84. <https://doi.org/10.1109/BTAS.2012.6374560>.

Zhang, K., Zuo, W., Chen, Y., Meng, D. and Zhang, L., 2017. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7), pp.3142–3155. <https://doi.org/10.1109/TIP.2017.2662206>.

APPENDICES

APPENDIX A: Images

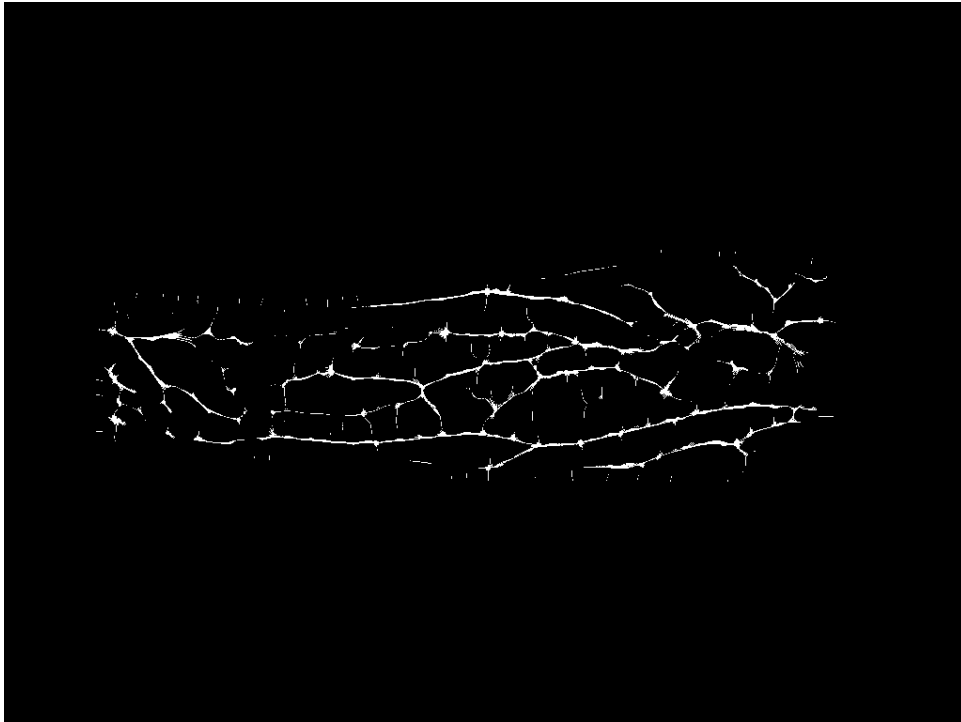


Image A-1: Segmented Veins Using Maximum Curvature.

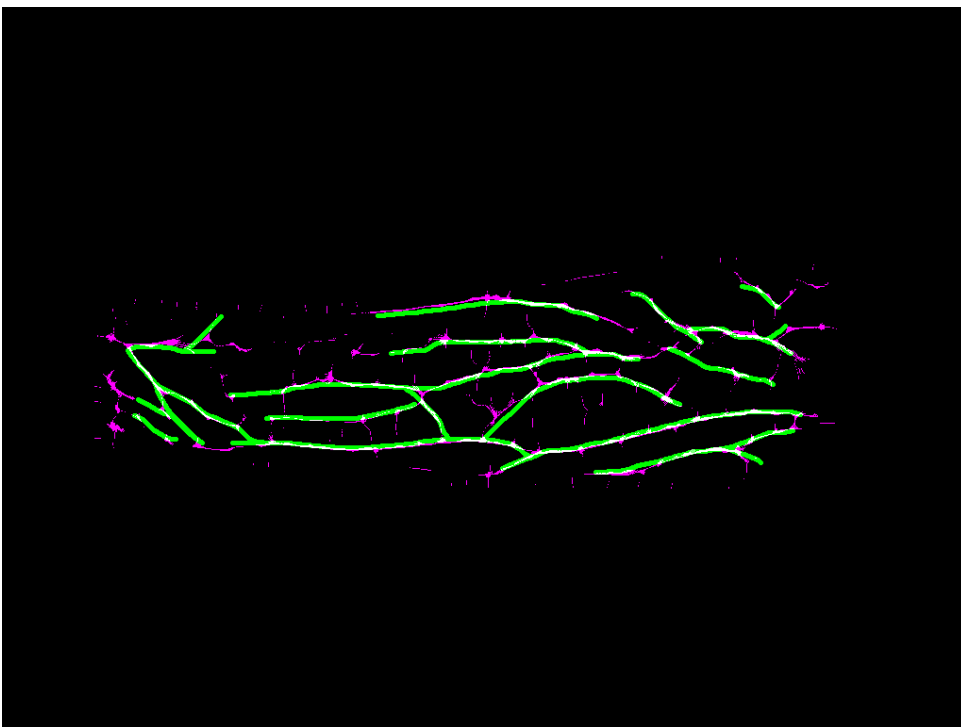


Image A-2: Overlaid of Segmented Veins Using Maximum Curvature (Purple) on Ground Truth (Green).

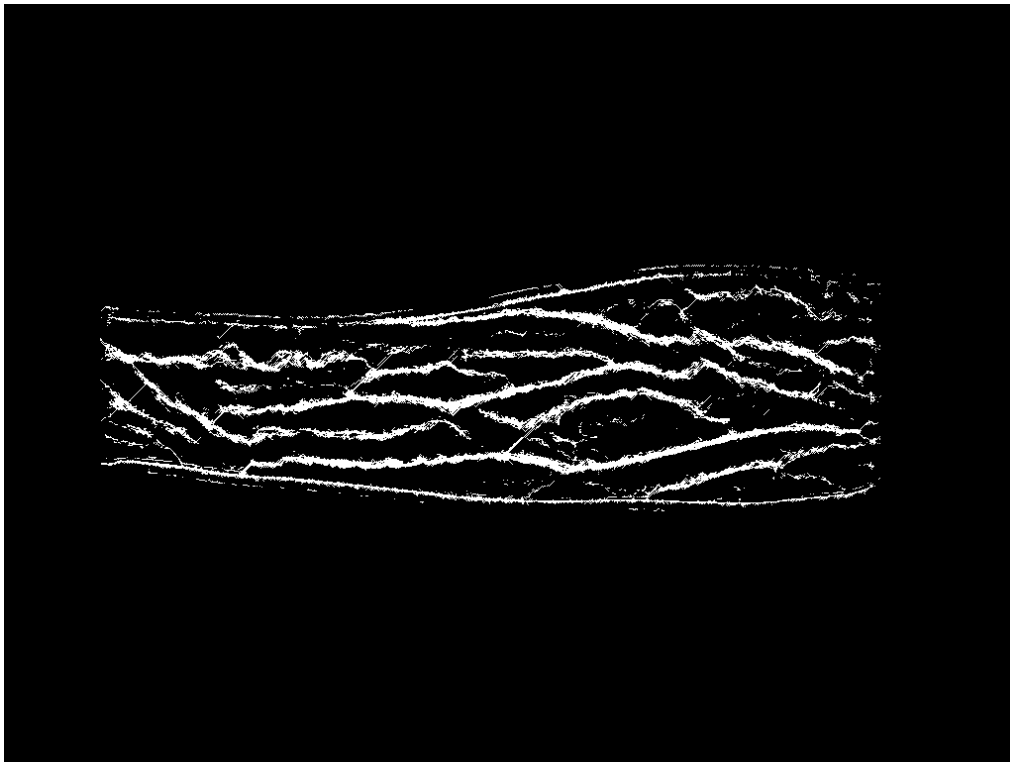


Image A-3: Segmented Veins Using Repeated Line Tracking.

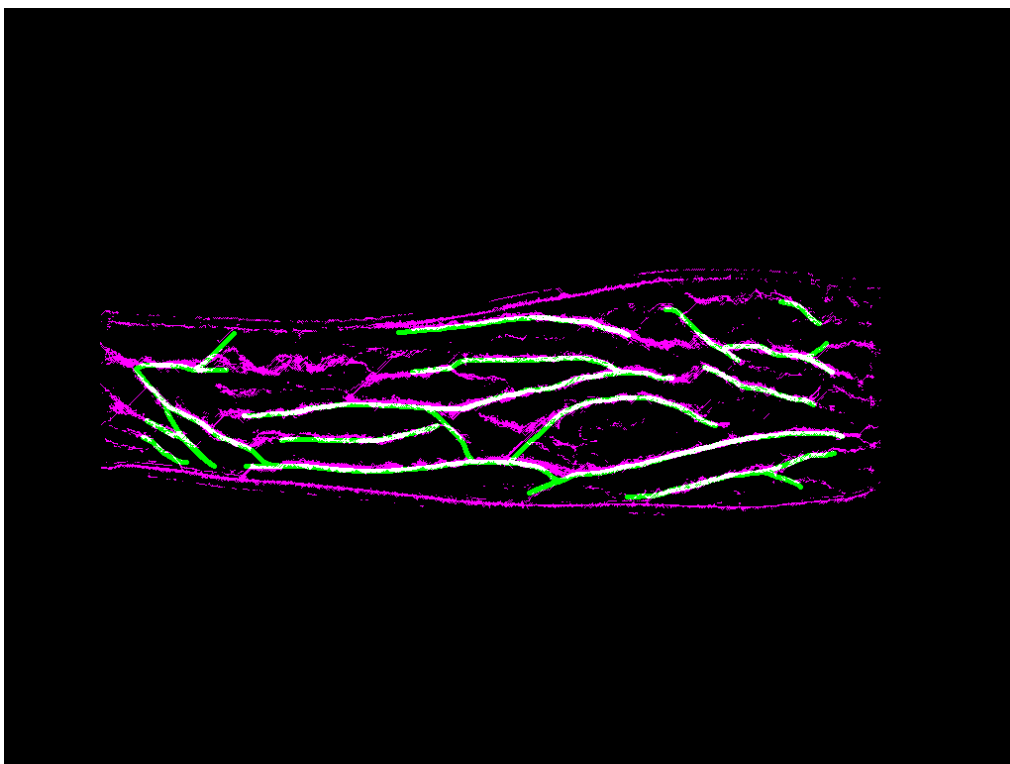


Image A-4: Overlaid of Segmented Veins Using Repeated Line Tracking (Purple) on Ground Truth (Green).

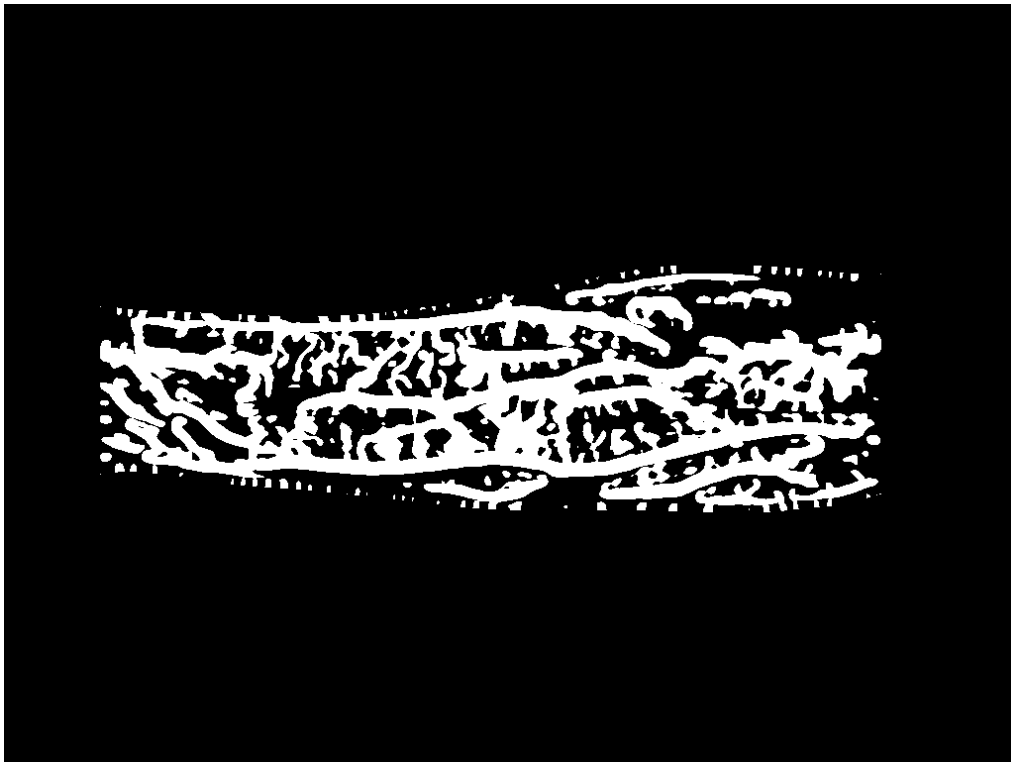


Image A-5: Segmented Veins Using Principal Curvature.

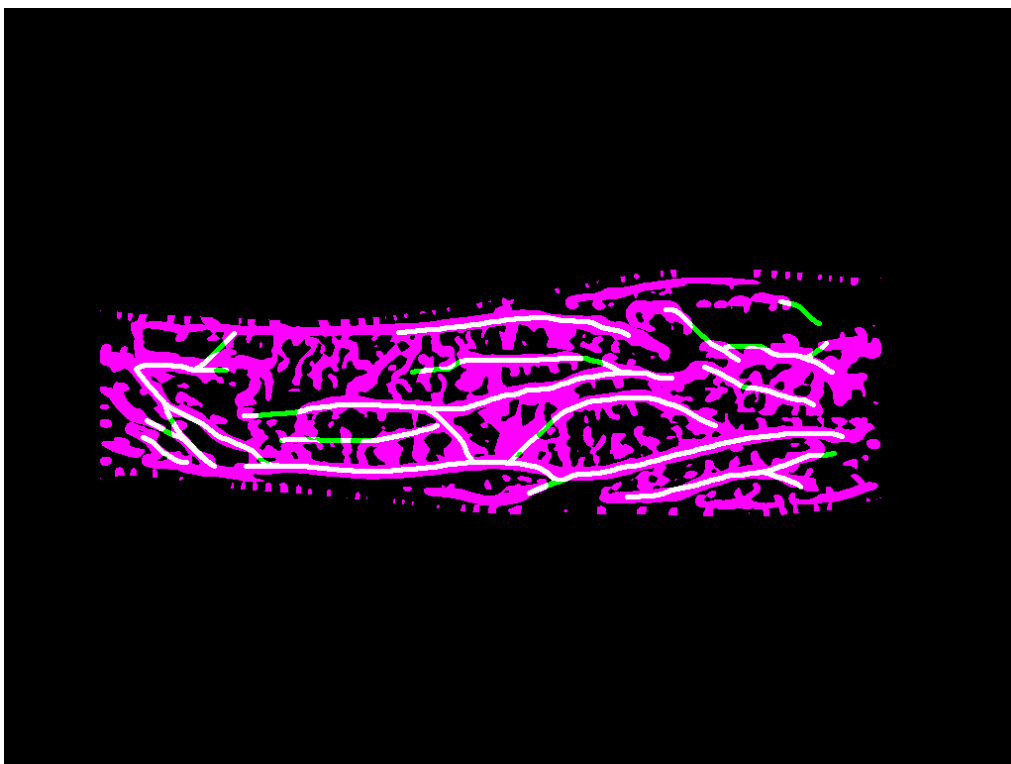


Image A-6: Overlaid of Segmented Veins Using Principal Curvature (Purple) on Ground Truth (Green).

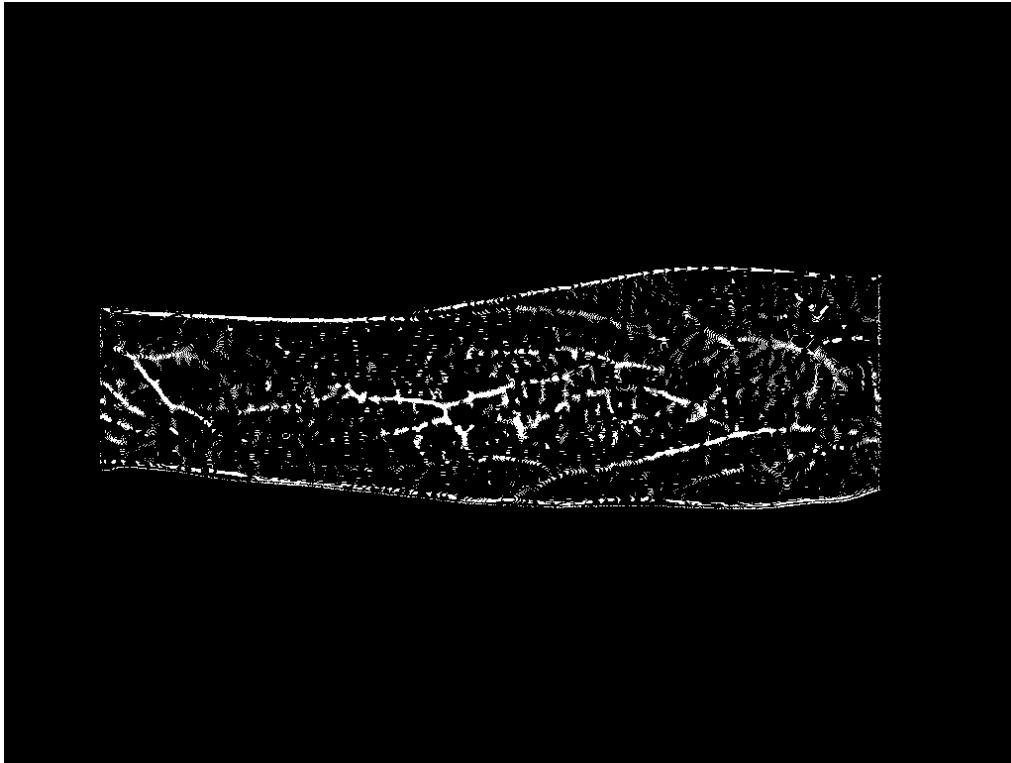


Image A-7: Segmented Veins Using Wide Line Tracking.

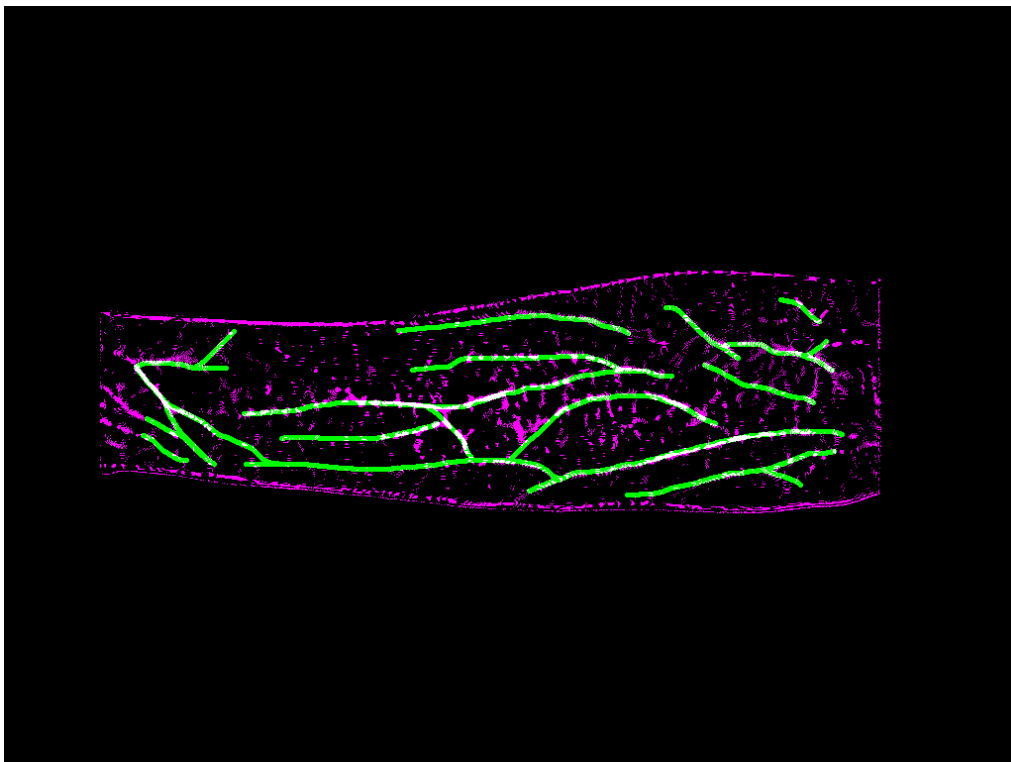


Image A-8: Overlaid of Segmented Veins Using Wide Line Tracking (Purple) on Ground Truth (Green).

The screenshot shows the MATLAB Editor window with the script 'MaxCurve_batch.m'. The script includes the following code:

```

1 - clear all
2 - clc
3
4 - addpath('..\Feature_extraction');
5 - addpath('..\Evaluation');
6
7
8 - datasets_location = 'D:\Datasets\NTU_v1\Preprocessed Forearm Region\*.bmp'; % folder in which you
9 - ds = imageDatastore(datasets_location); % Creates a datastore for all images in your folder
10
11 - mask_location = 'D:\Datasets\NTU_v1\Forearm Mask\';
12 - GT_location = 'D:\Datasets\NTU_v1\Ground truth\';
13
14 % Initialize the cell
15 - Accuracy = [];
16 - Specificity = [];
17 - Sensitivity = [];
18 - Precision = [];

```

The Command Window shows the following output:

```

New to MATLAB? See resources for Getting Started.
Handled image 244
Handled image 245
Handled image 246
Handled image 247
Handled image 248
Handled image 249
Handled image 250
Elapsed time is 2216.149841 seconds.
fx >>

```

Image A-9: Time Taken to Perform Maximum Curvature on 250 Images.

The screenshot shows the MATLAB Editor window with the script 'RepeatedLine_batch.m'. The script includes the following code:

```

12 - GT_location = 'D:\Datasets\NTU_v1\Ground truth\';
13
14 % Initialize the cell
15 - Accuracy = [];
16 - Specificity = [];
17 - Sensitivity = [];
18 - Precision = [];
19 - MCC = [];
20 - Dice = [];
21 - Jaccard = [];
22
23 - k = 0;
24
25 - seg_gt_location = 'D:\Program\Segmentation_NTU_v1\Results\Repeated Line Tracking\DetectionVSGround\';
26 - seg_location = 'D:\Program\Segmentation_NTU_v1\Results\Repeated Line Tracking\VeinsDetected\';
27
28 - tic
29 - while hasdata(ds)

```

The Command Window shows the following output:

```

New to MATLAB? See resources for Getting Started.
Handled image 244
Handled image 245
Handled image 246
Handled image 247
Handled image 248
Handled image 249
Handled image 250
Elapsed time is 709.220276 seconds.
fx >>

```

Image A-10: Time Taken to Perform Repeated Line Tracking on 250 Images.


```

Editor - D:\Program\Segmentation_NTU_v1\WideLine_batch.m
+7 WideLine_batch.m x PrincipalCurve_unit.m x PrincipalCurve_batch.m x pre_processing_badge.m x pre_processing_unit.m x +
10
11 mask_location = 'D:\Datasets\NTU_v1\Forearm Mask\';
12 GT_location = 'D:\Datasets\NTU_v1\Ground truth\';
13
14 % Initialize the cell
15 Accuracy = [];
16 Specificity = [];
17 Sensitivity = [];
18 Precision = [];
19 MCC = [];
20 Dice = [];
21 Jaccard = [];
22
23 k = 0;
24
25 seg_gt_location = 'D:\Program\Segmentation_NTU_v1\Results\WideLine Tracking\DetectionVSGround\';
26 seg_location = 'D:\Program\Segmentation_NTU_v1\Results\WideLine Tracking\VeinsDetected\';
27
Command Window
New to MATLAB? See resources for Getting Started.
Handled image 244
Handled image 245
Handled image 246
Handled image 247
Handled image 248
Handled image 249
Handled image 250
Elapsed time is 476.366948 seconds.
fx >>

```

Image A-11: Time Taken to Perform Wide Line Detector on 250 Images.

```

Editor - D:\Program\Segmentation_NTU_v1\PrincipalCurve_batch.m
+7 WideLine_batch.m x PrincipalCurve_unit.m x PrincipalCurve_batch.m x pre_processing_badge.m x pre_processing_unit.m x +
1 clear all
2 clc
3
4 addpath('.\Feature_extraction');
5 addpath('.\Evaluation');
6
7
8 datasets_location = 'D:\Datasets\NTU_v1\Preprocessed Forearm Region\*.bmp'; % folder in which you
9 ds = imageDatastore(datasets_location); % Creates a datastore for all images in your folder
10
11 mask_location = 'D:\Datasets\NTU_v1\Forearm Mask\';
12 GT_location = 'D:\Datasets\NTU_v1\Ground truth\';
13
14 % Initialize the cell
15 Accuracy = [];
16 Specificity = [];
17 Sensitivity = [];
18 Precision = [];
19
Command Window
New to MATLAB? See resources for Getting Started.
Handled image 244
Handled image 245
Handled image 246
Handled image 247
Handled image 248
Handled image 249
Handled image 250
Elapsed time is 147.325375 seconds.
fx >>

```

Image A-12: Time Taken to Perform Principal Curvature on 250 Images.

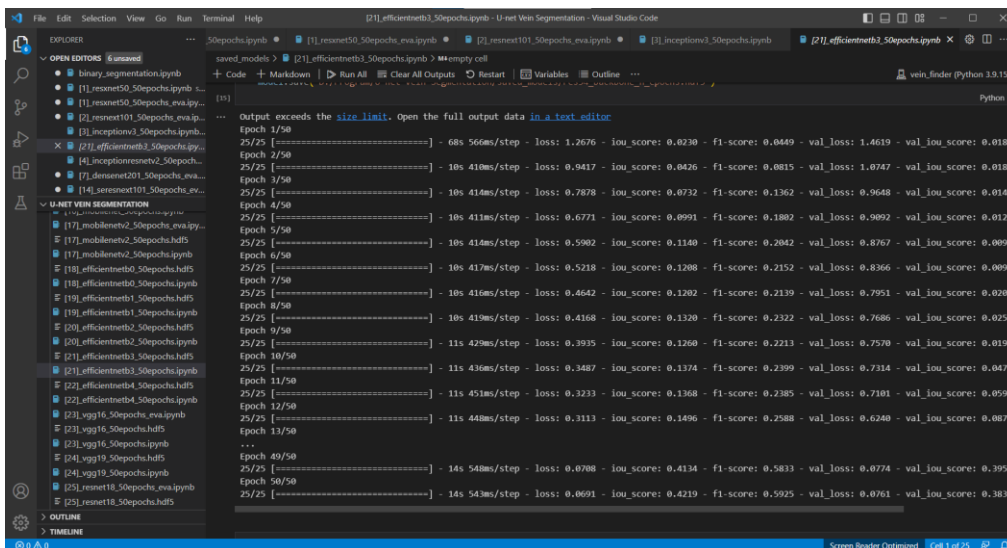


Image A-13: Training process of U-net using EfficientNetB3 as backbone.

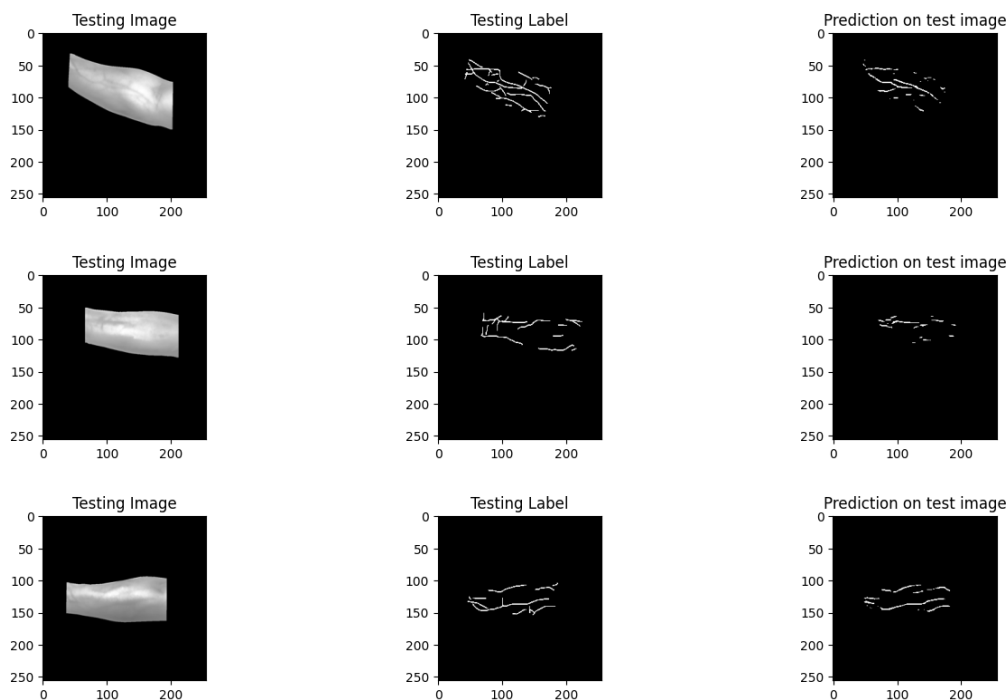


Image A-14: Sample images of testing result using U-net with EfficientNetB3 as backbone.

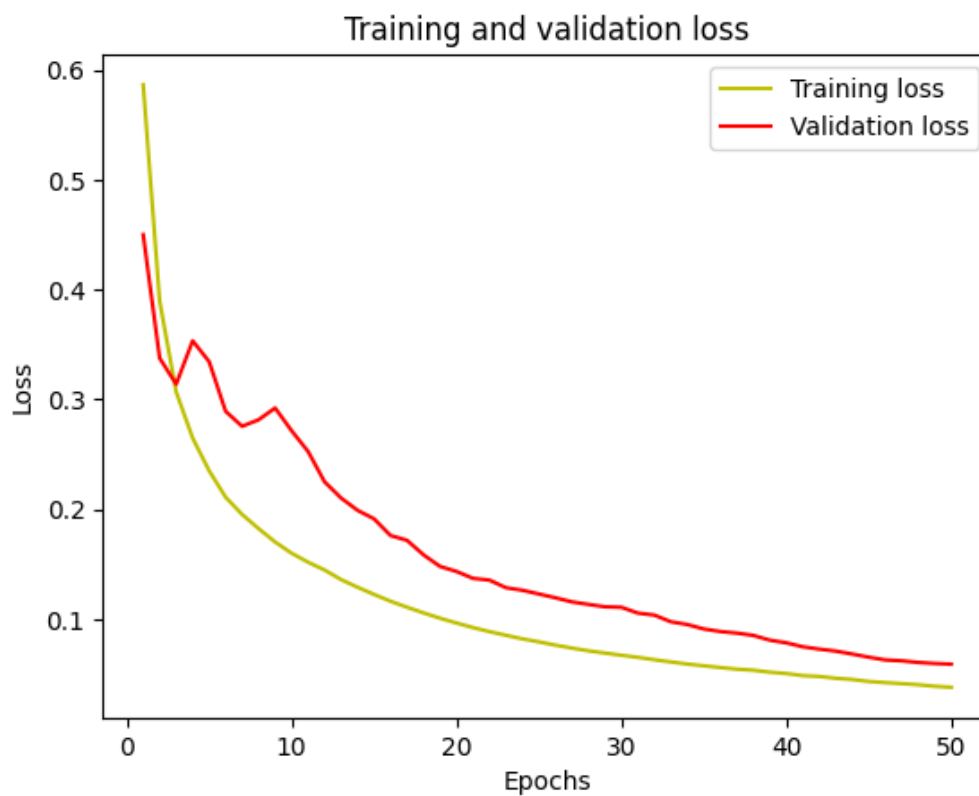


Image A-15: Training and Validation Loss of ResNeXt-50.

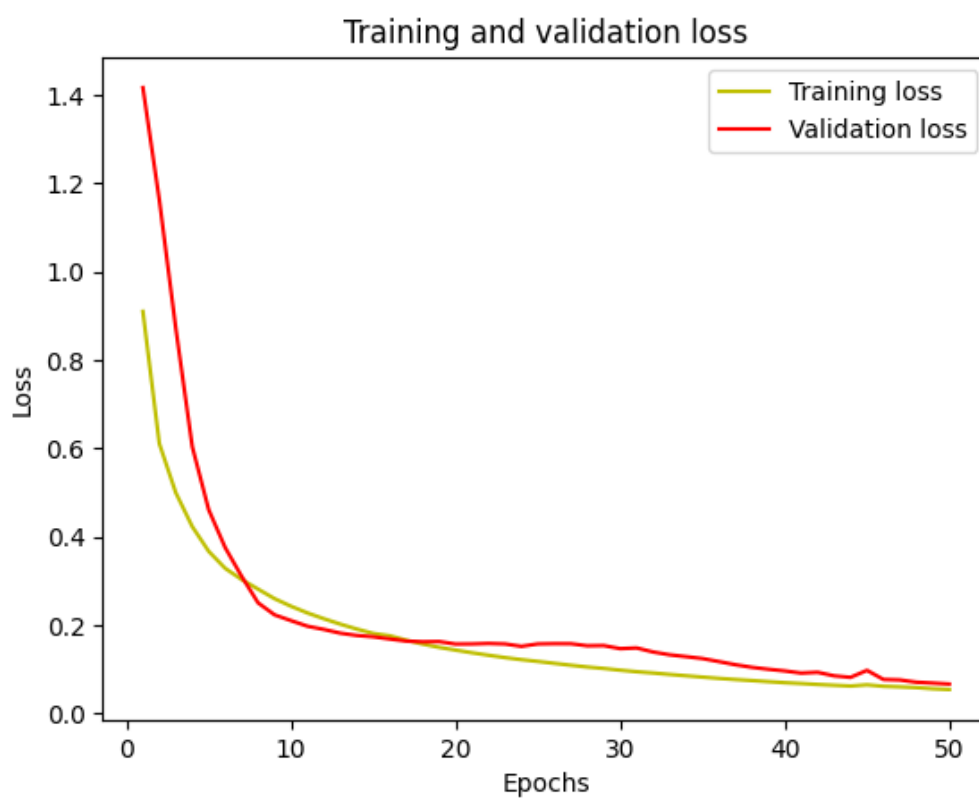


Image A-16: Training and Validation Loss of ResNeXt-101.

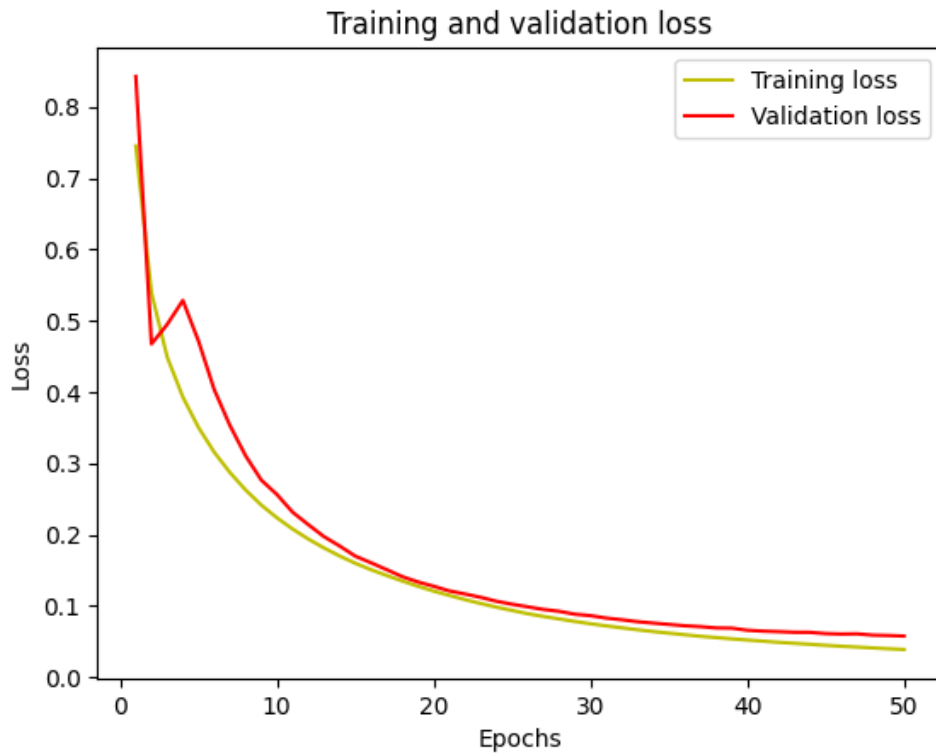


Image A-17: Training and Validation Loss of Inception-v3.

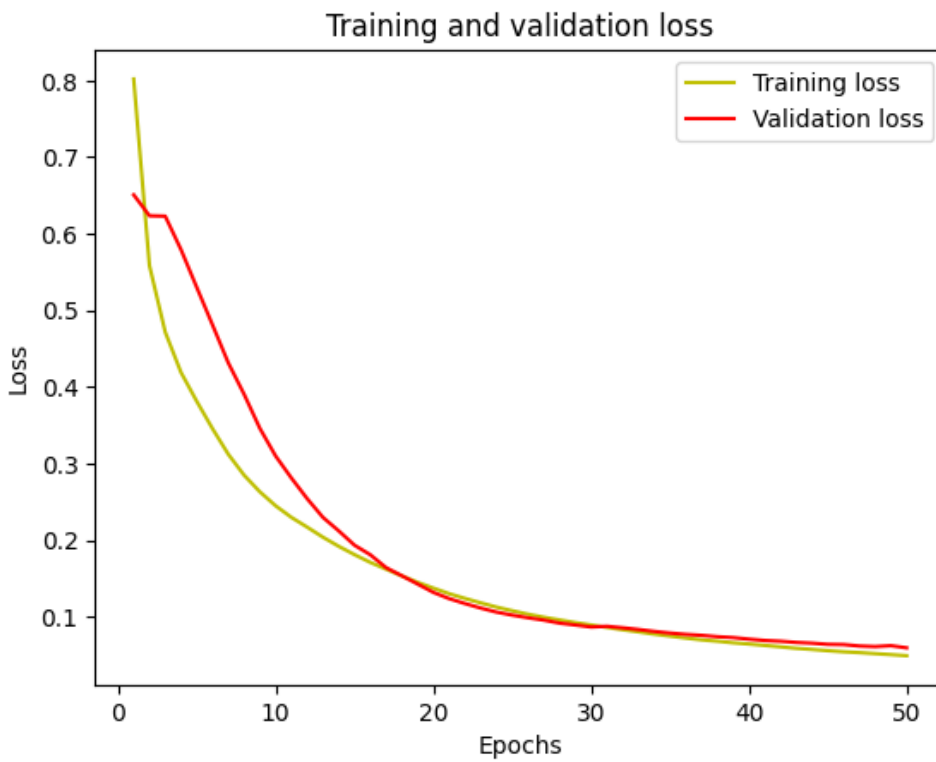


Image A-18: Training and Validation Loss of Inception-ResNet-v2.

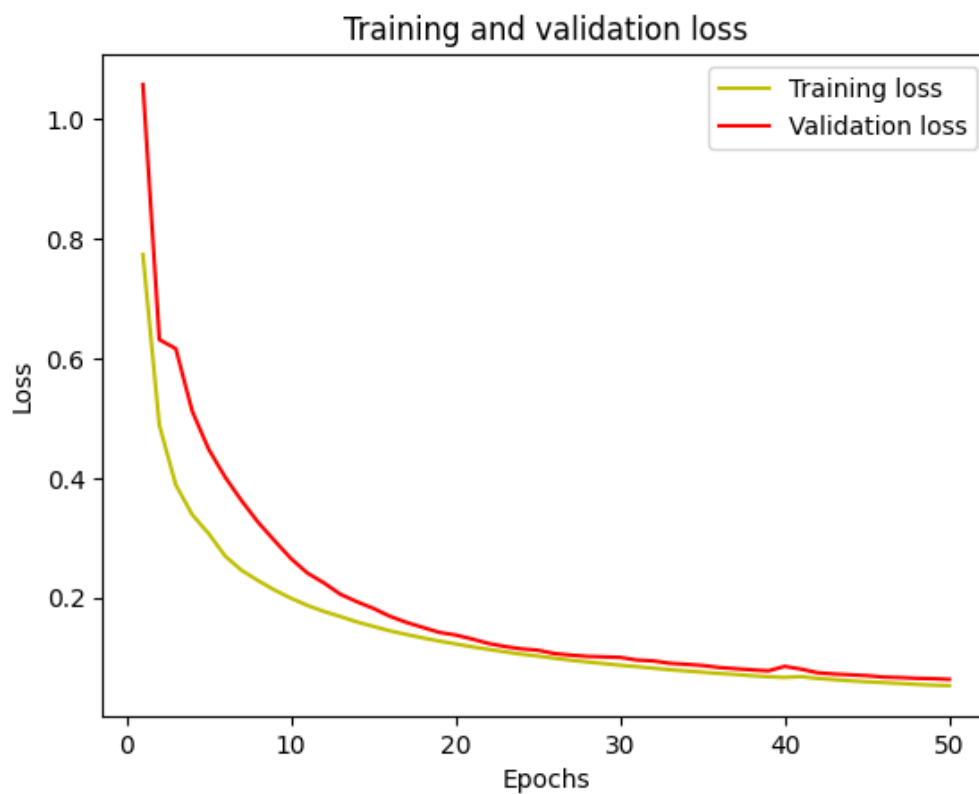


Image A-19: Training and Validation Loss of DenseNet121.

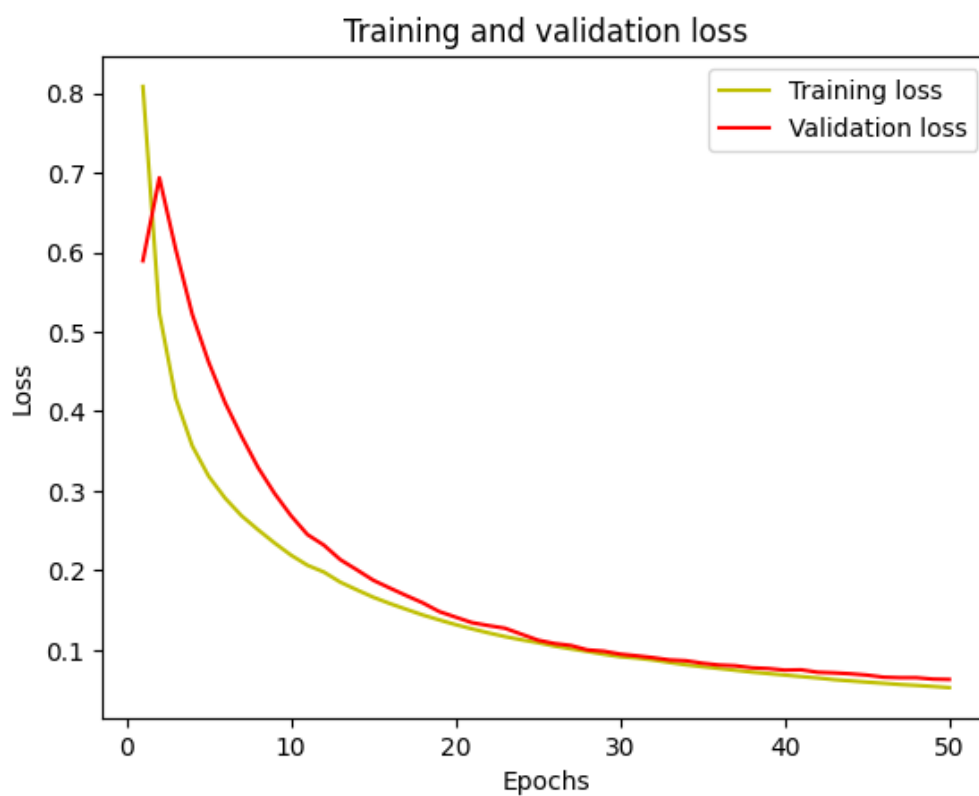


Image A-20: Training and Validation Loss of DenseNet169.

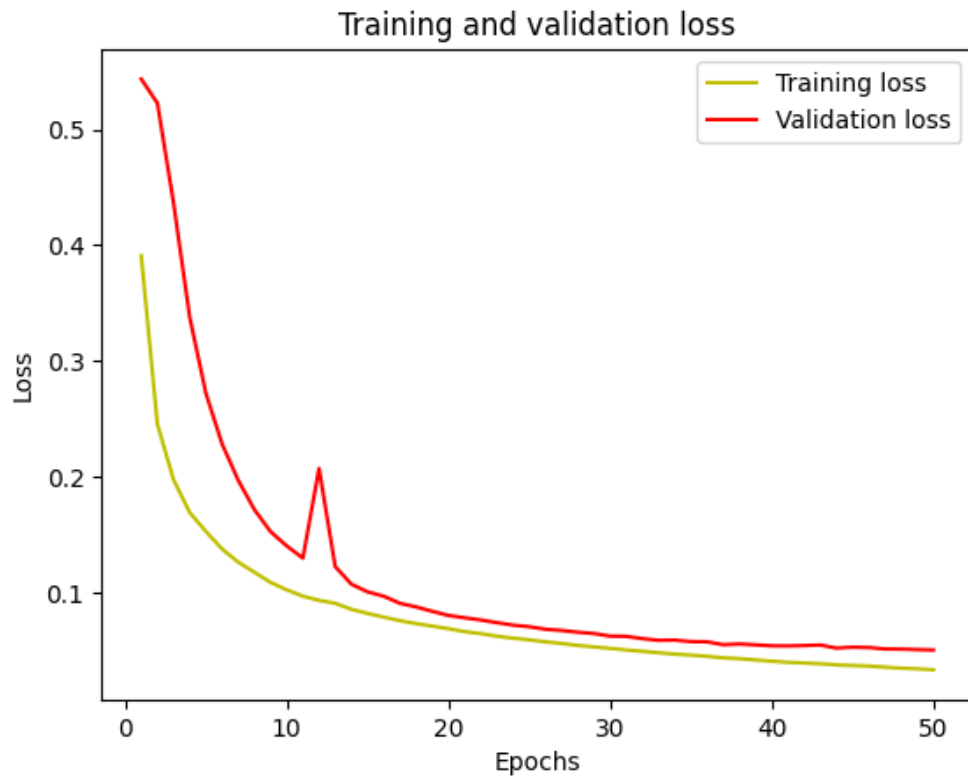


Image A-21: Training and Validation Loss of DenseNet201.

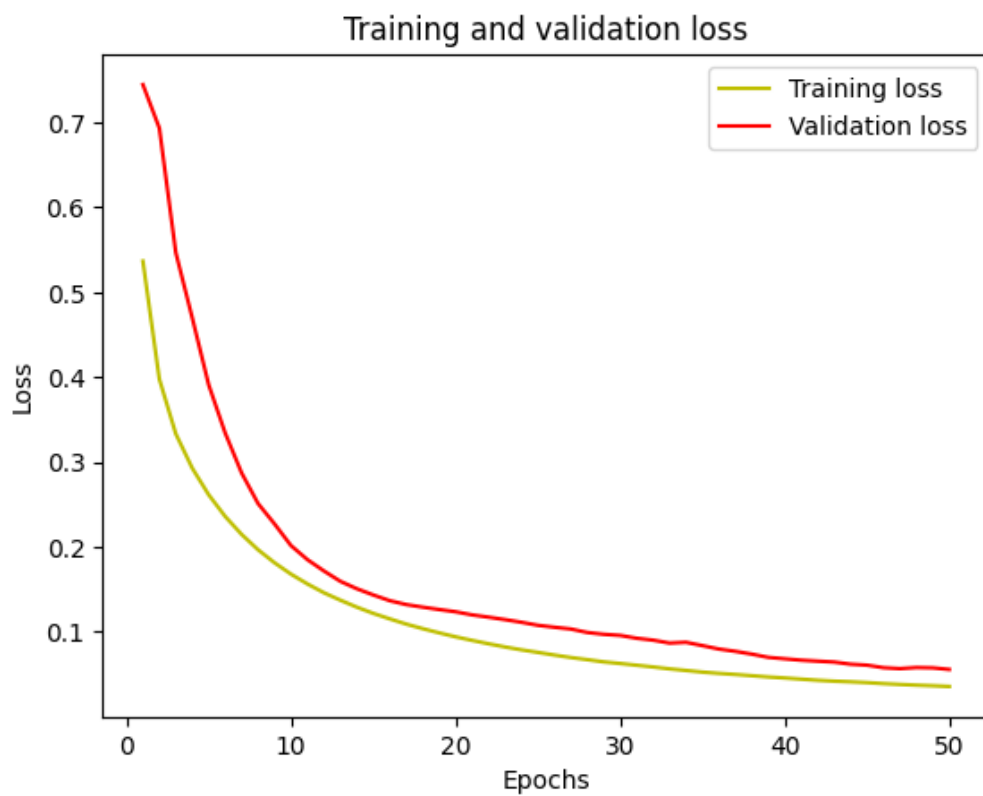


Image A-21: Training and Validation Loss of SeresNet18.

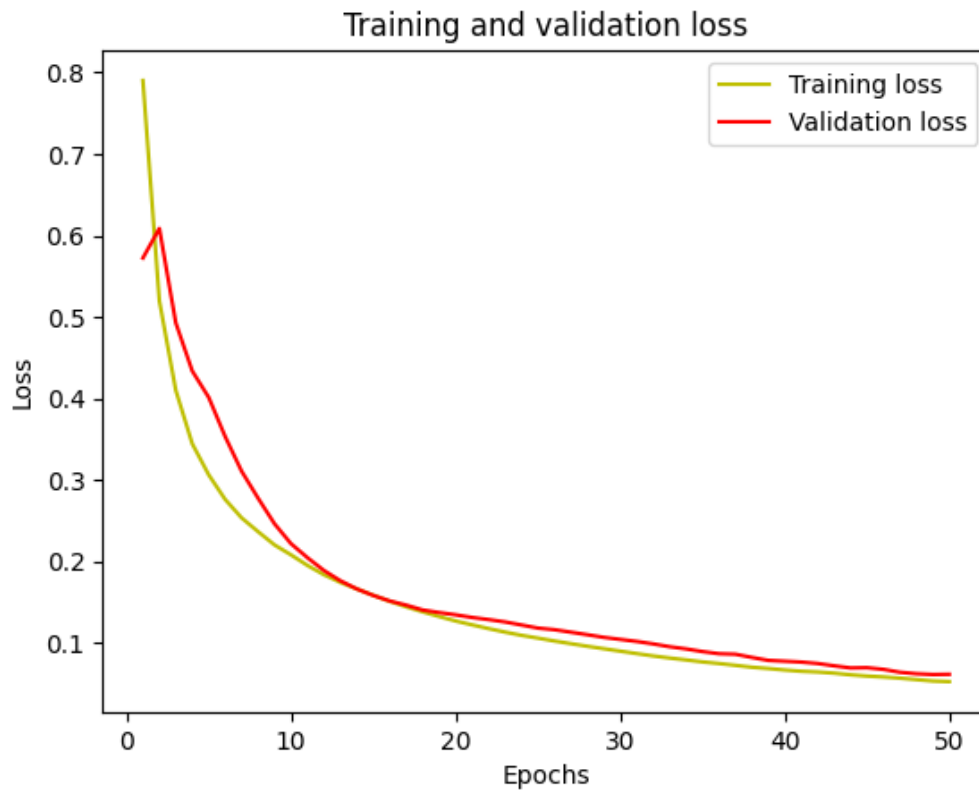


Image A-22: Training and Validation Loss of SeresNet34.

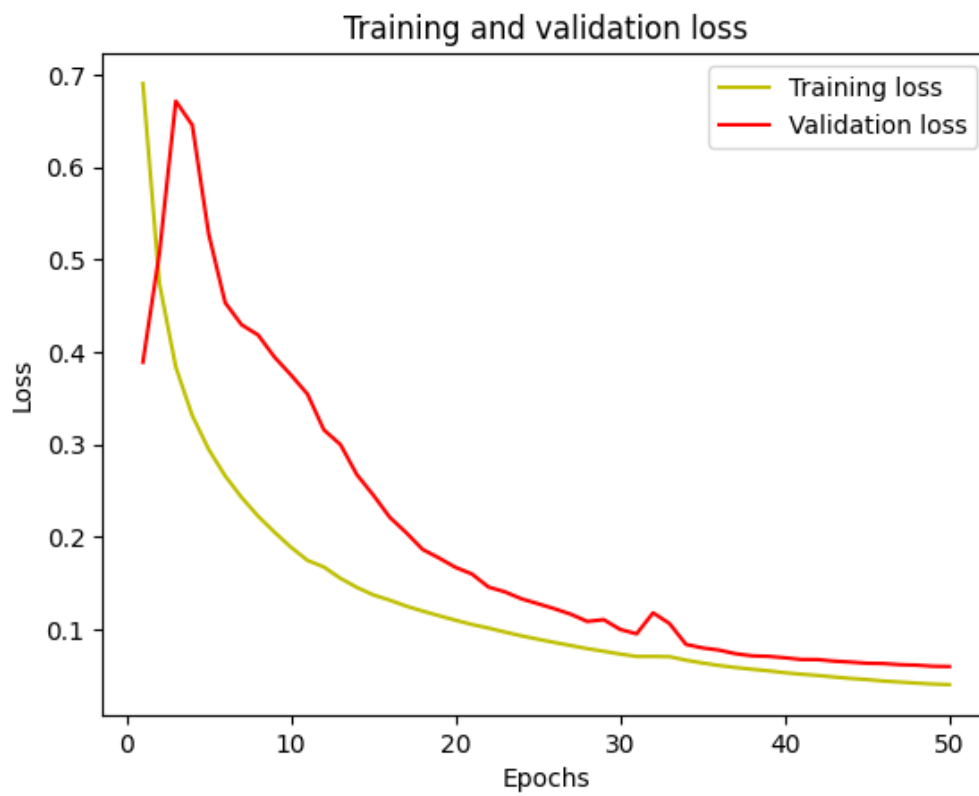


Image A-23: Training and Validation Loss of SeresNet50.

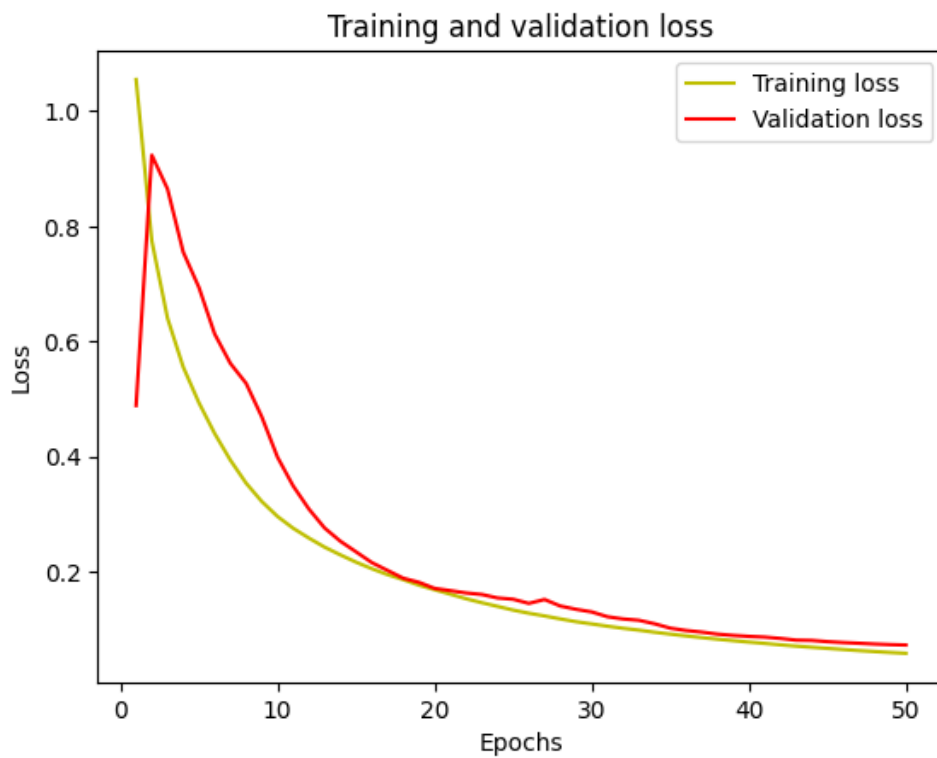


Image A-24: Training and Validation Loss of SEResNet101.

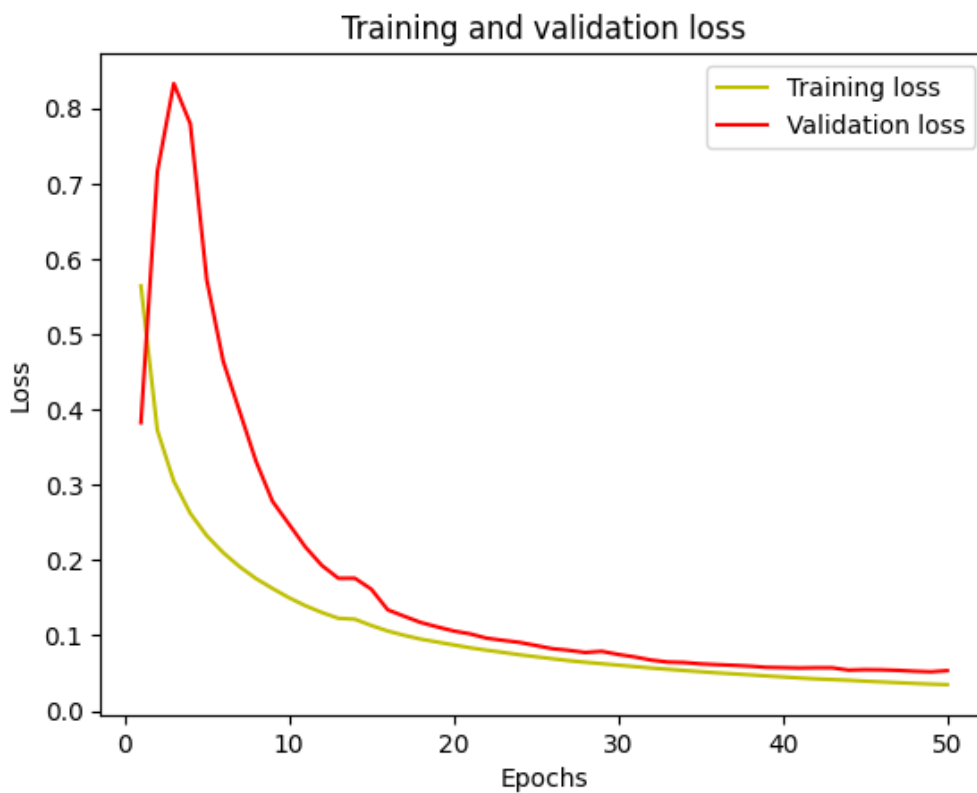


Image A-25: Training and Validation Loss of SEResNet152.

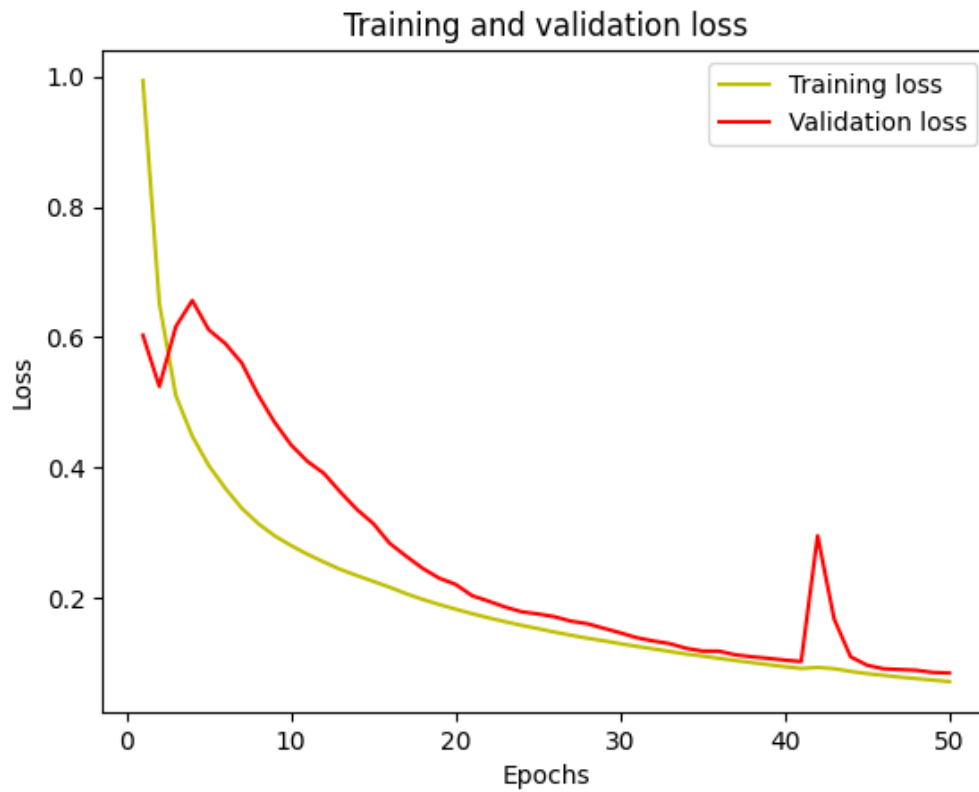


Image A-26: Training and Validation Loss of SEResNext50.

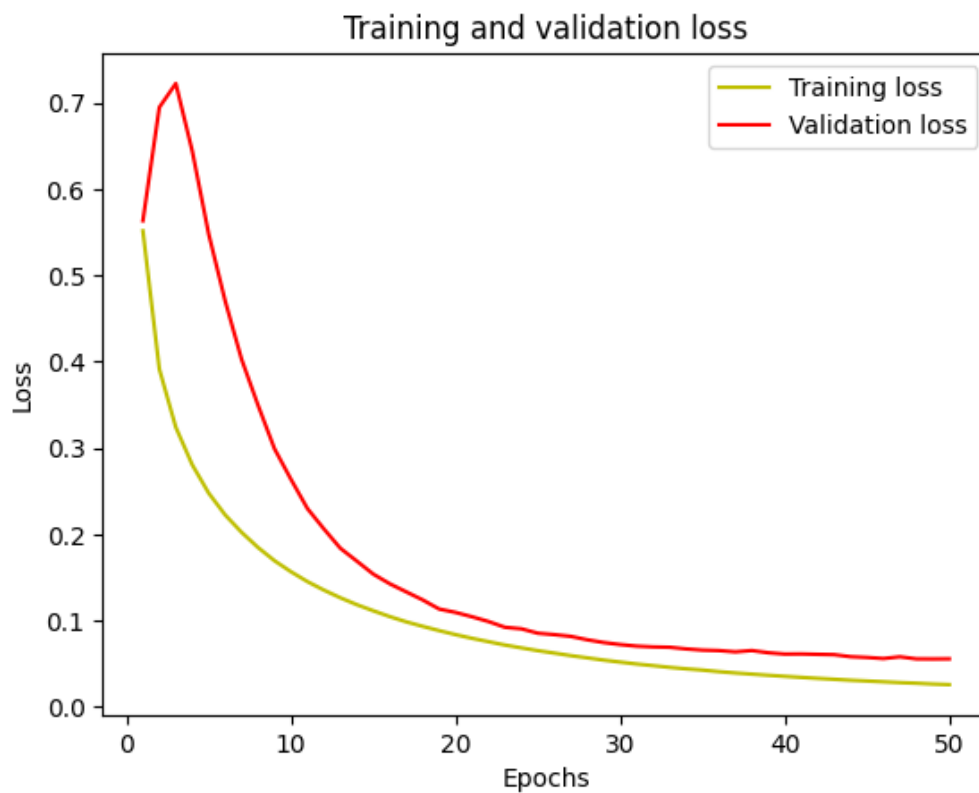


Image A-27: Training and Validation Loss of SEResNext101.

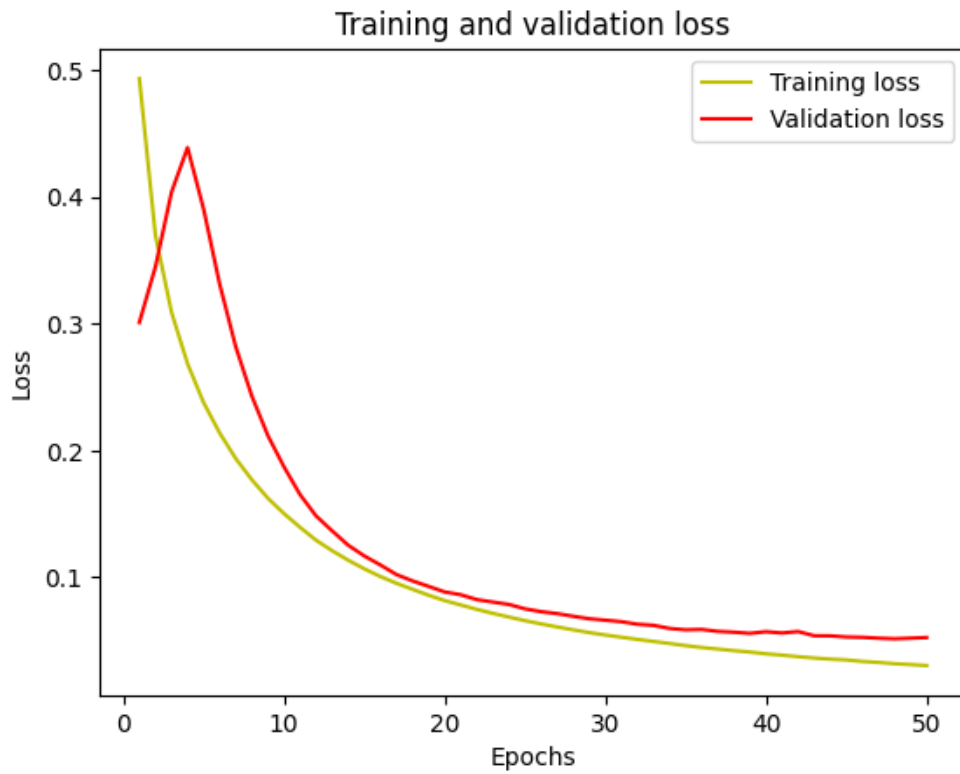


Image A-28: Training and Validation Loss of MobileNet.

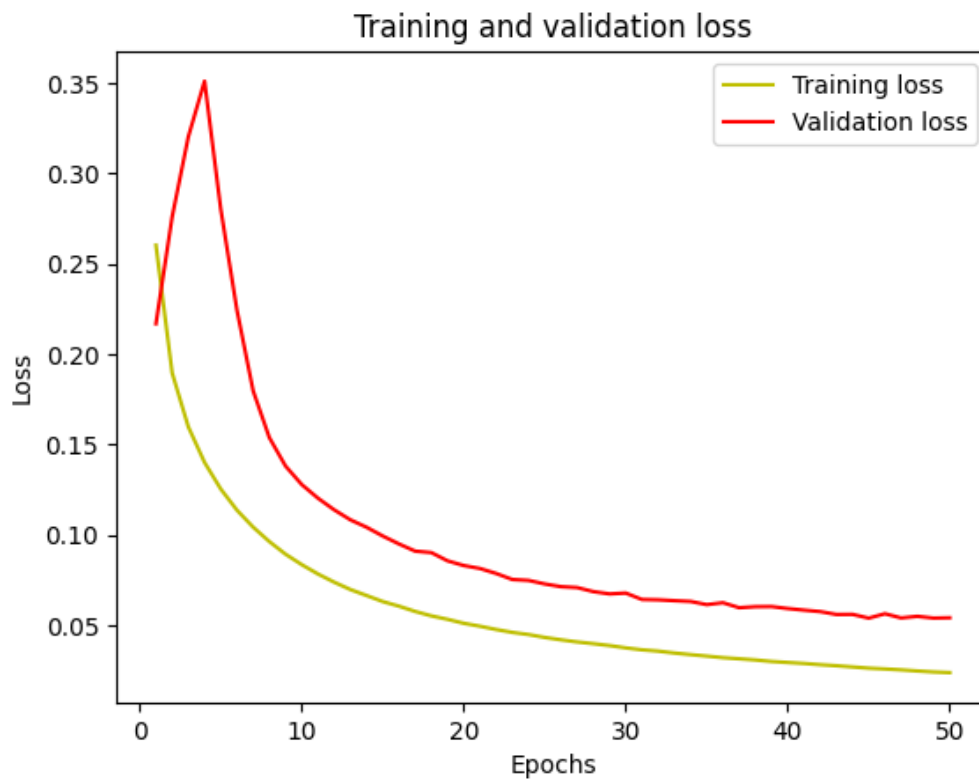


Image A-29: Training and Validation Loss of MobileNet v2.

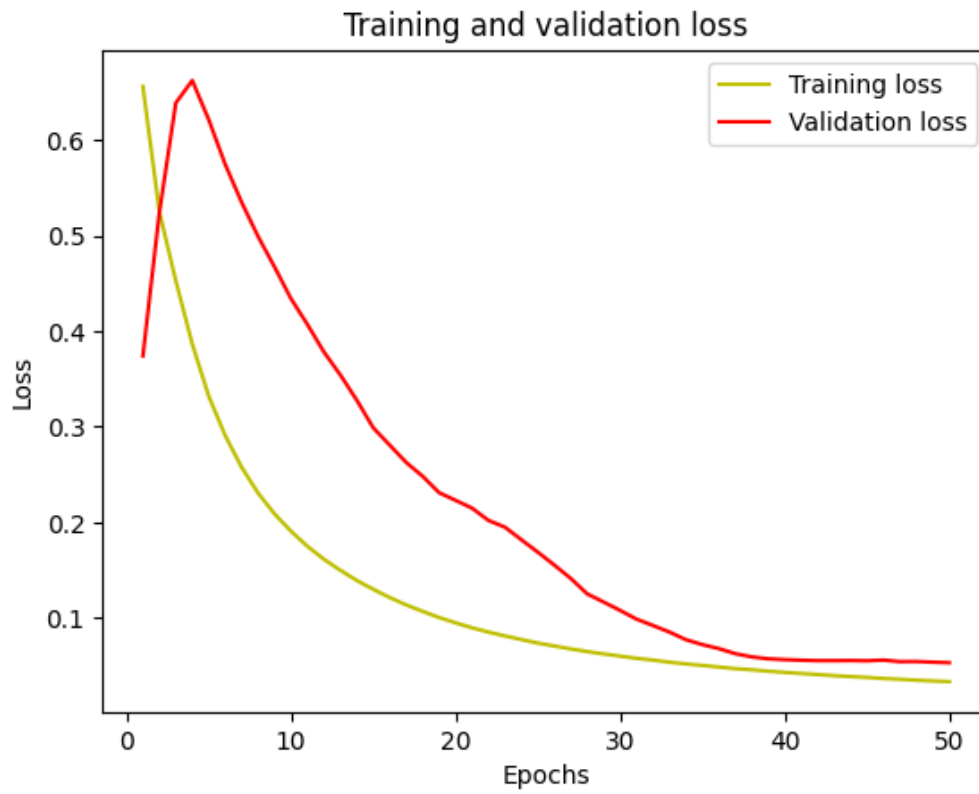


Image A-30: Training and Validation Loss of EfficientNetB0.

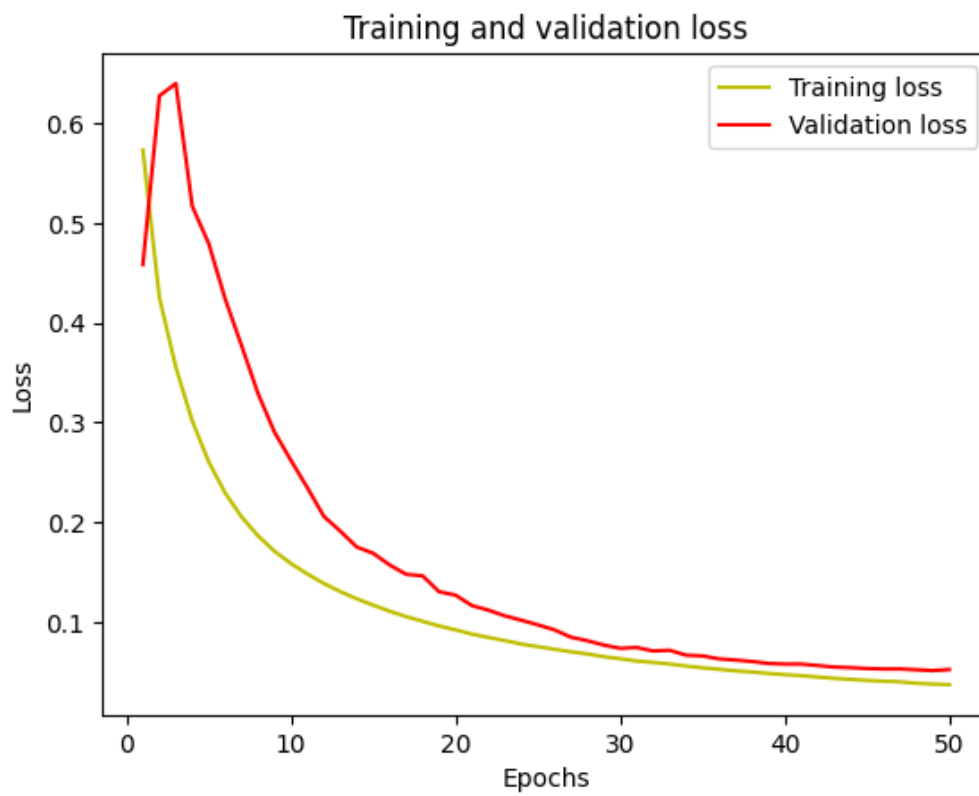


Image A-31: Training and Validation Loss of EfficientNetB1.

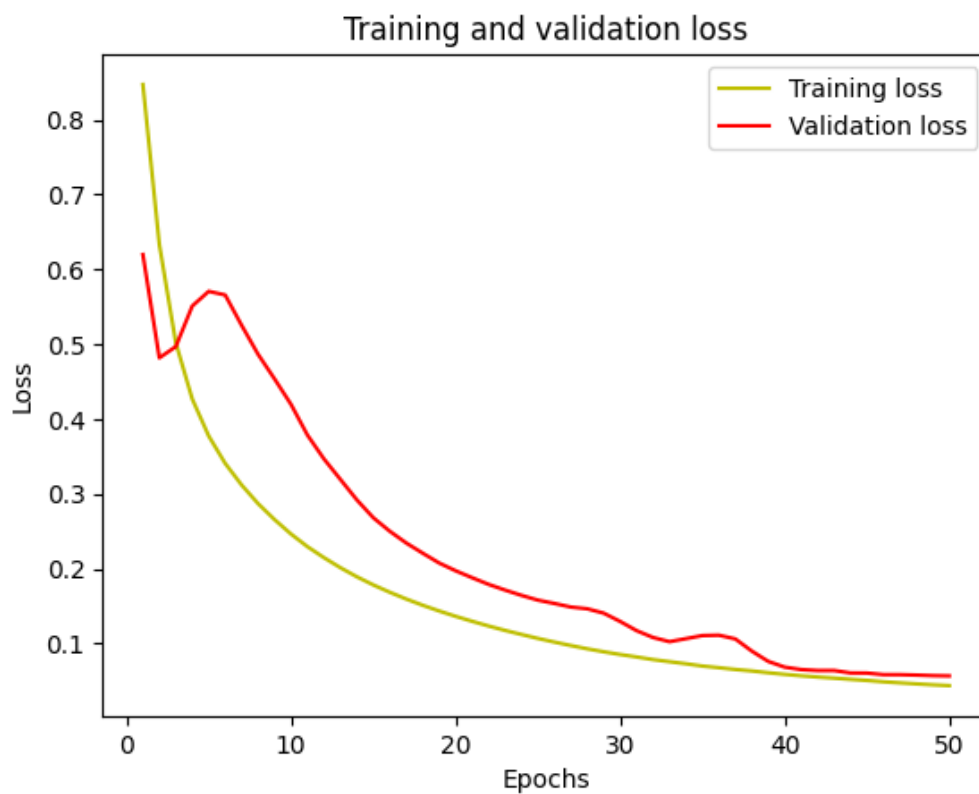


Image A-32: Training and Validation Loss of EfficientNetB2.

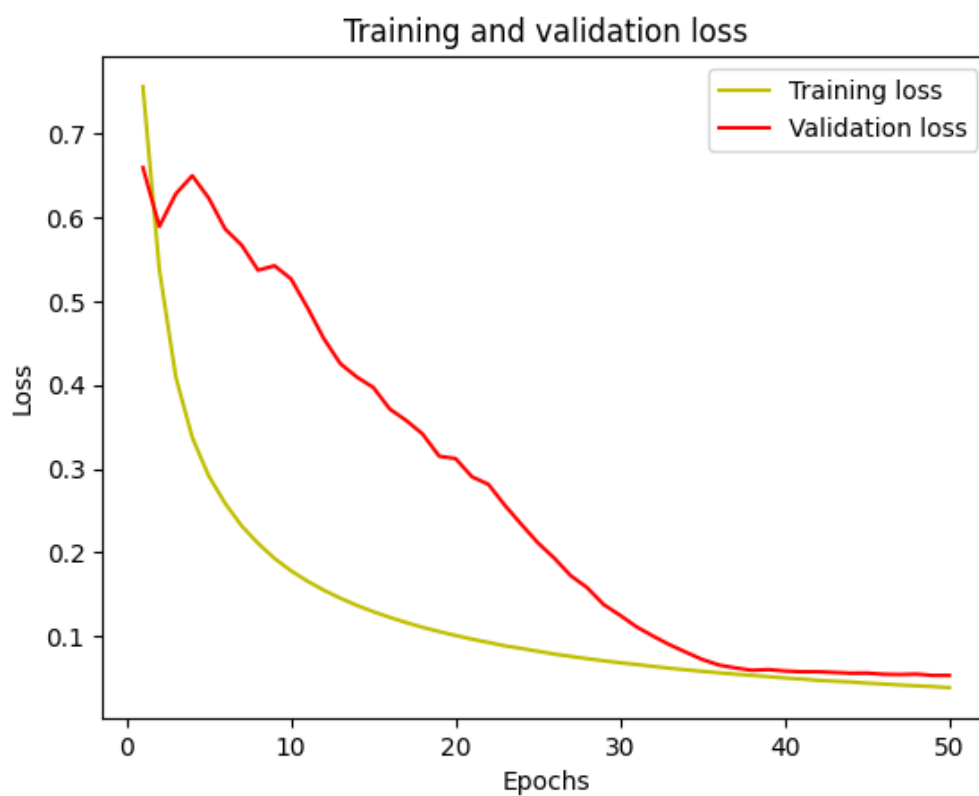


Image A-33: Training and Validation Loss of EfficientNetB3.

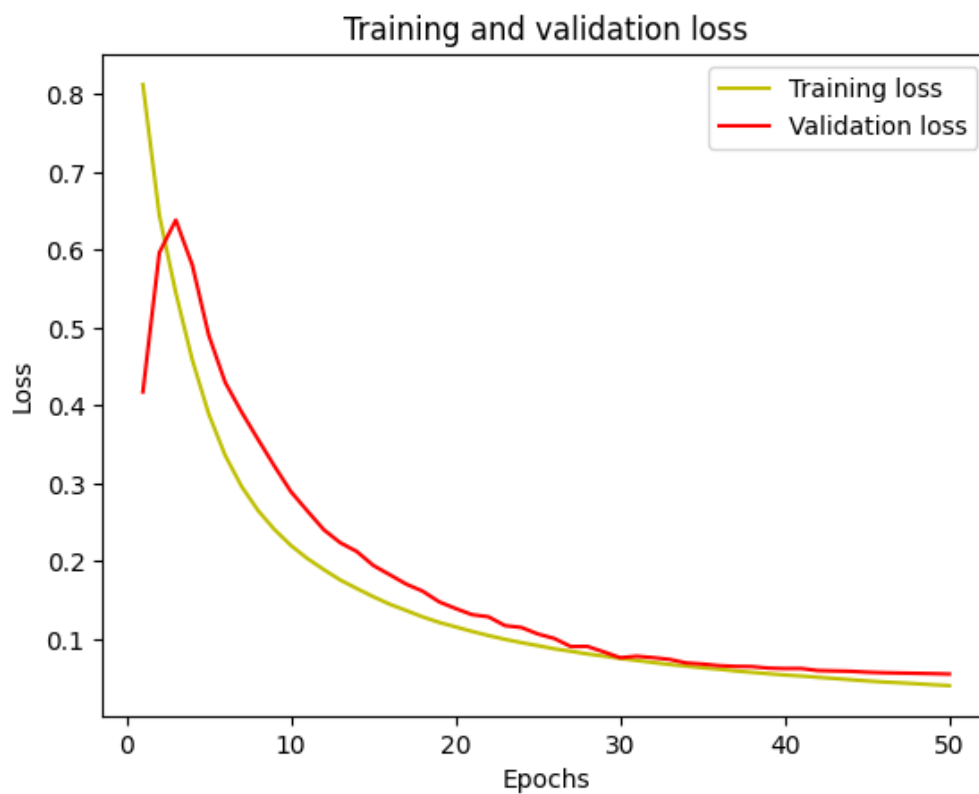


Image A-34: Training and Validation Loss of EfficientNetB4.

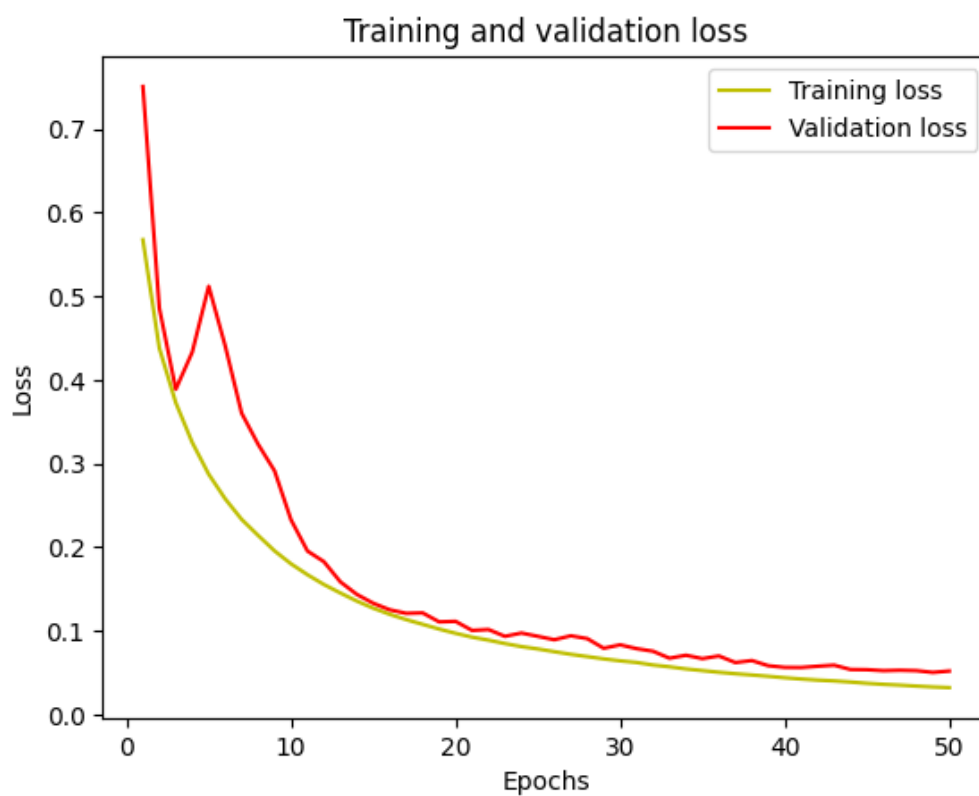


Image A-35: Training and Validation Loss of VGG16.

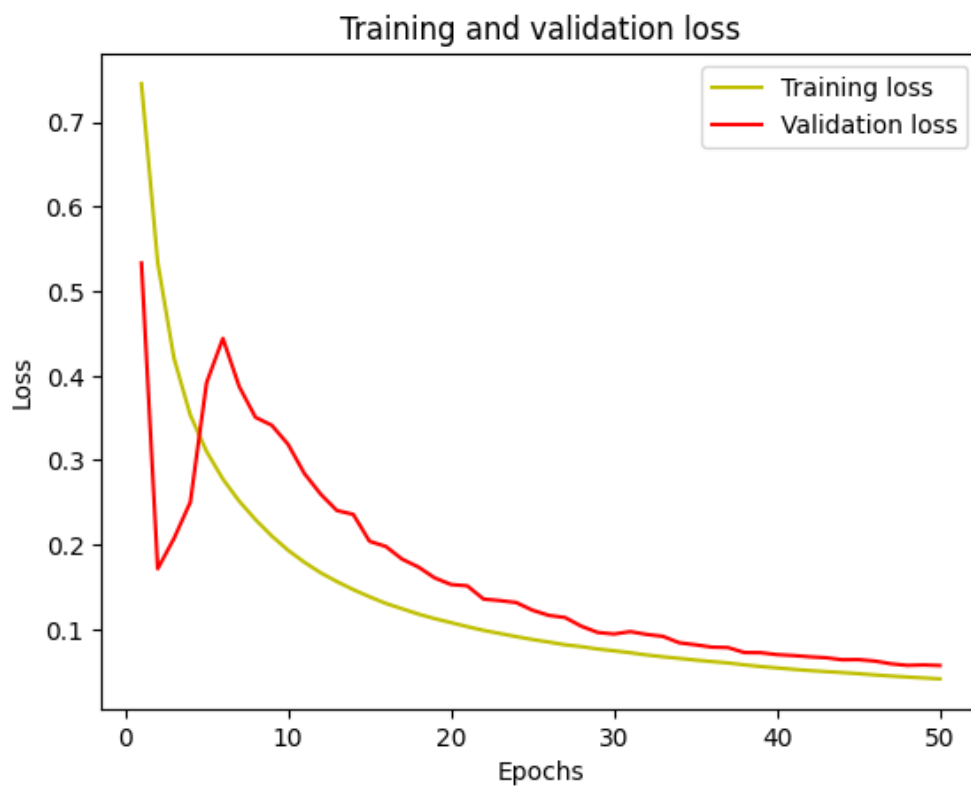


Image A-36: Training and Validation Loss of VGG19.

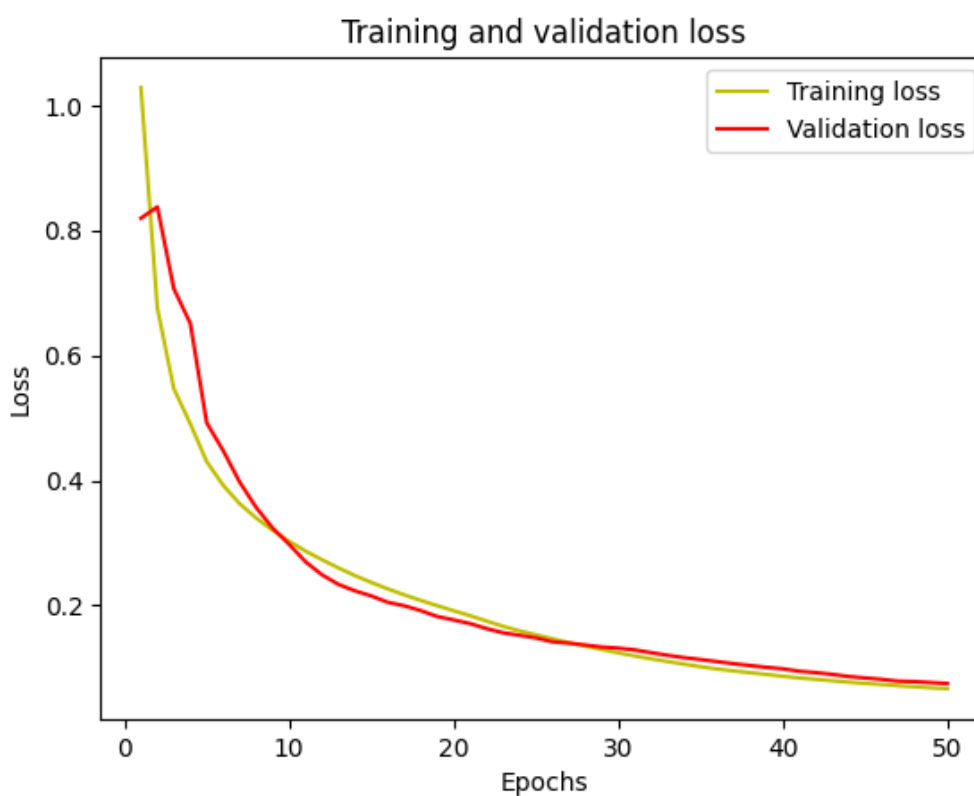


Image A-37: Training and Validation Loss of ResNet18.

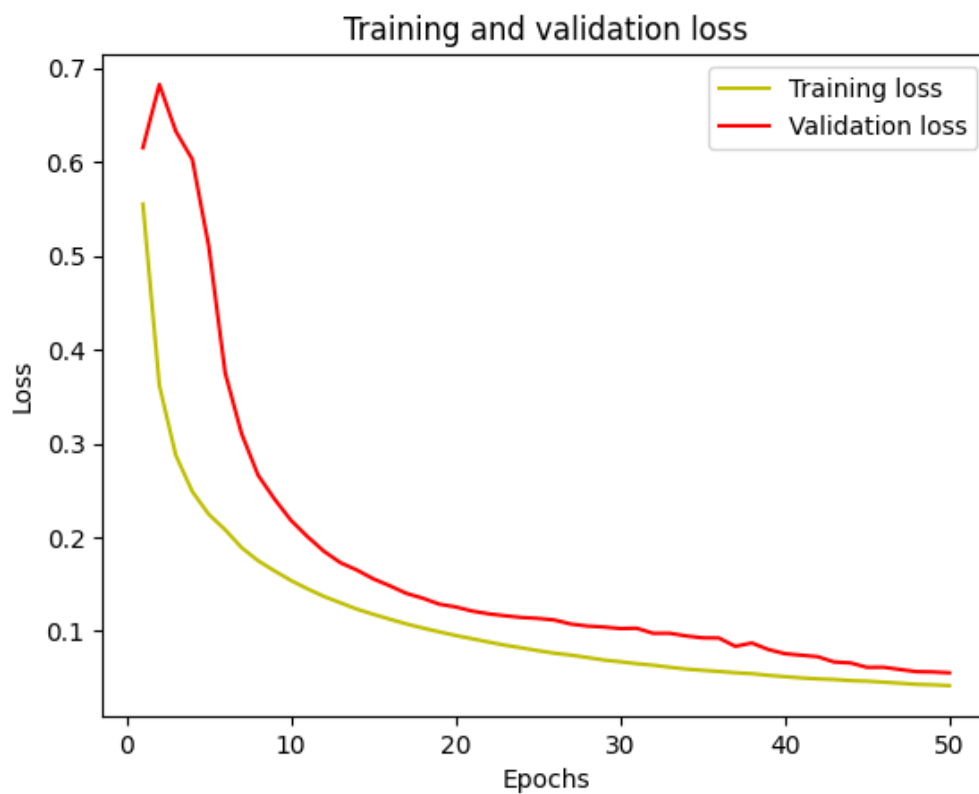


Image A-38: Training and Validation Loss of ResNet34.

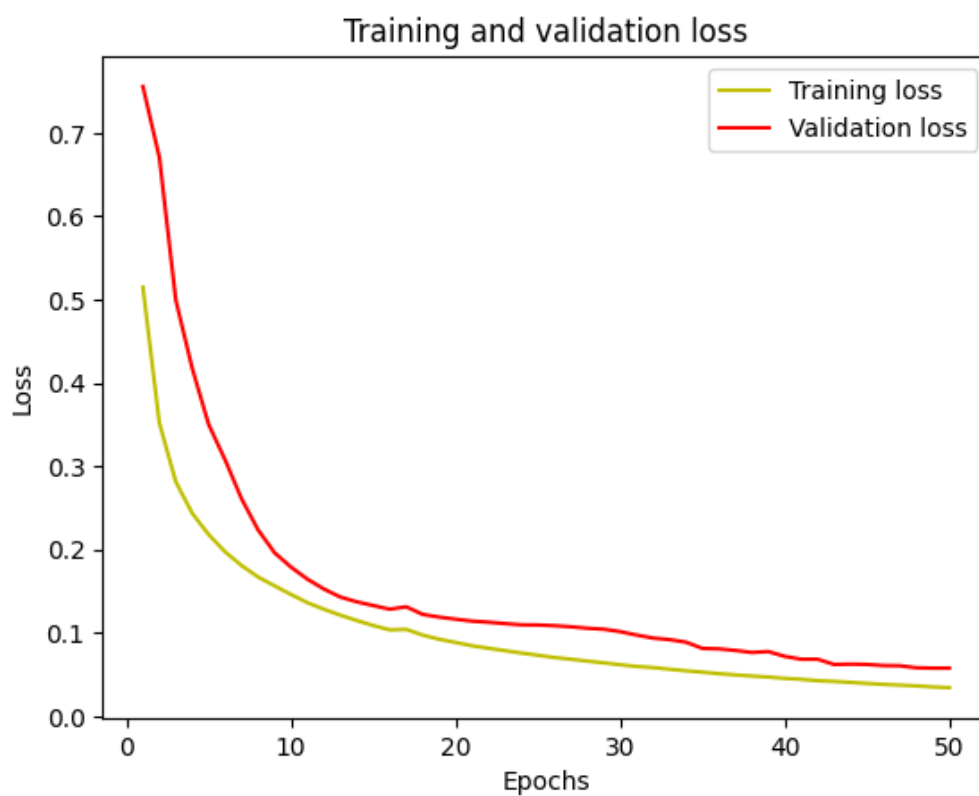


Image A-39: Training and Validation Loss of ResNet50.

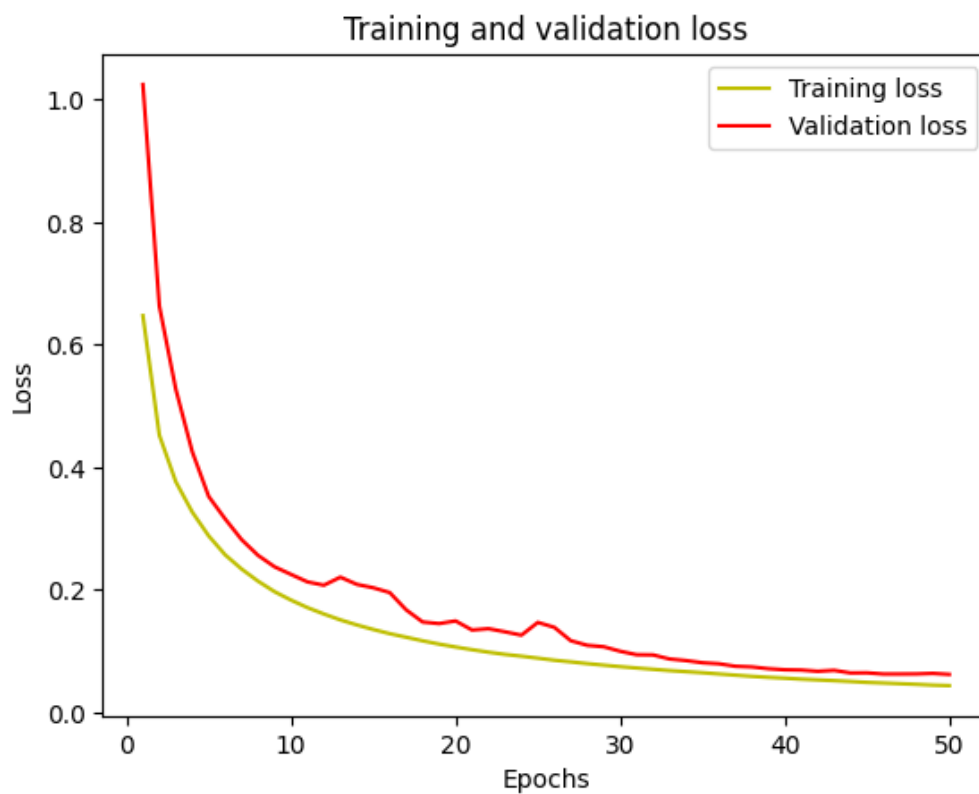


Image A-40: Training and Validation Loss of ResNet101.

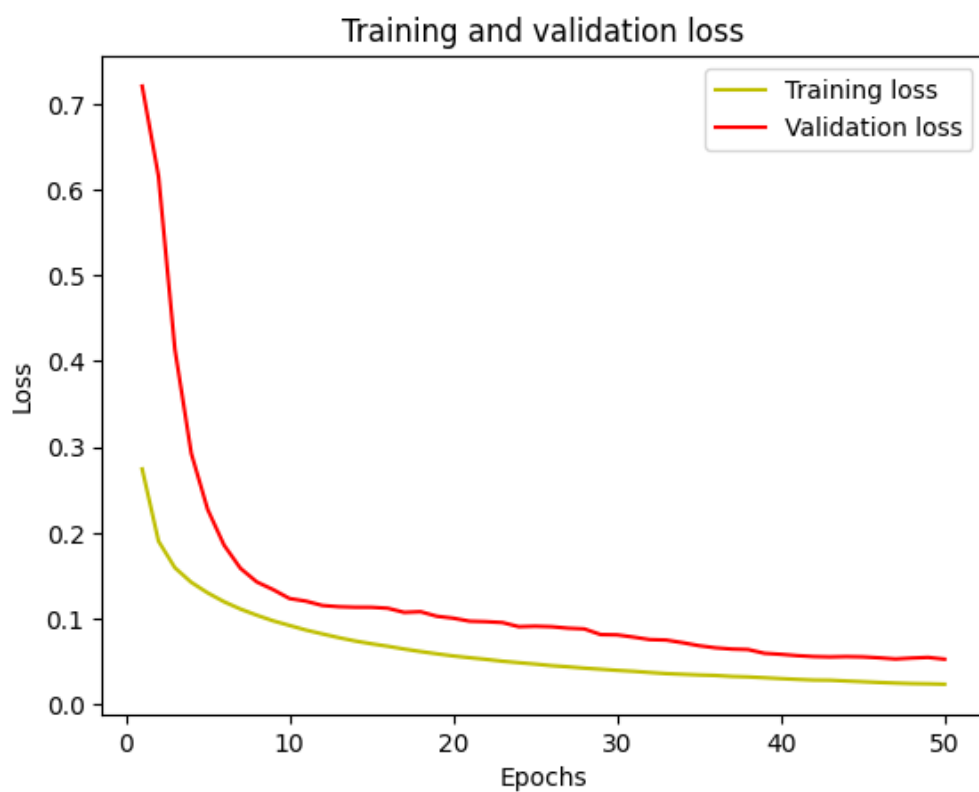


Image A-41: Training and Validation Loss of ResNet-152.