An Improved Semi-Supervised Learning Framework for Image Semantic Segmentation

Nusrat Jahan

Electrical and Computer Engineering

Lakehead University, Thunder Bay, Ontario

A thesis submitted to Lakehead University in partial fulfillment
of the requirements for the Master of Science degree
in the Electrical and Computer Engineering

Thesis Committee Members

The members listed below served on the Examining Committee for this thesis:

Supervisor:	Dr. Thangarajah Akilan Department of Software Engineering.
Internal Committee Member:	Dr. Hassan Naser Department of Software Engineering.
External Committee Member:	Dr. Saad Bin Ahmed Department of Computer Science.
Session Chair:	Dr. Yushi Zhou Department of Electrical and Computer Engineering.

Declaration of Co-Authorship / Previous Publications

I. Co-Authorship Declaration

I hereby declare that this dissertation includes material resulting from joint research, as outlined below: It incorporates outcomes of research publications conducted under the supervision of Dr. Thangarajah Akilan, with collaborations from Dr. Garima Bajwa (Chapter 3), Tharrengini Suresh (Chapter 4), and Dr. Thanh Minh Nguyen (Chapter 5). In all instances, I am the primary author, and I carried out the key tasks, including main idea generation, experimental designs, data analysis, interpretation, and writing, while the co-author's contributions are limited to proofreading and reviewing the technical content of the research papers.

I am fully aware of the Lakehead University Policy on Authorship, and I certify that I have properly acknowledged the contributions of other researchers to this dissertation. Additionally, I have obtained permission from each co-author of the respective conference publication declared in Section on page iii to include the respective publications' content in this dissertation.

With the above qualifications, I certify that this dissertation and the research it encompasses are my original work.

II. Declaration of Previous Publications

This thesis incorporates the content of three original research papers, all of which have been previously published or presented at IEEE conferences. The details are as follows:

Thesis chapter	Publication title/full citation	Status
Chapter 3	**N. Jahan**, G. Bajwa, and T. Akilan, "Federated learning-assisted self-supervised cnn for monkeypox diagnosis," in *2023 IEEE Western New York Image and Signal Processing Workshop*, 2023, pp. 1–5.	Published
Chapter 4	**N. Jahan**, T. Akilan and T. Suresh, "Improving Pavement Crack Segmentation Using Attention Mechanism and Self-gated Activation," in *2024 IEEE Canadian Conference on Electrical and Computer Engineering*, 2024.	Presented, awaiting publication
Chapter 5	**N. Jahan**, T. Akilan and T-M. Nguyen, "Improved Semi-supervised Attention GAN for Semantic Segmentation," in *2024 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, 2024.	Presented, awaiting publication

III. General

I declare that, to the best of my knowledge, my thesis does not infringe upon anyone's copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act. I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office. This thesis has not been submitted for a higher degree to any other University or Institution.

Acknowledgements

I would like to express my deepest gratitude to my thesis supervisor, Dr. Thangarajah Akilan, whose guidance, encouragement, and support were invaluable throughout my research journey. His expertise, insights, and constructive comments greatly refined my ideas and elevated the quality of my work.

I am also deeply grateful to the members of my thesis committee, Dr. Hassan Naser, Dr. Saad Bin Ahmed, and Dr. Yushi Zhou, for their valuable feedback and suggestions.

Finally, I am indebted to my family for their unwavering love, support, and encouragement. Their belief in me provided me the strength to persevere through the challenges I faced.

Dedication

This thesis is dedicated to my dear parents.

Abstract

Traditional supervised learning methods depend heavily on labeled data, which is both costly and time-intensive to acquire. Self-supervised learning approaches present a promising alternative to supervised learning, enabling the utilization of unlabeled data. Thus, this research aims to build an advanced semi-supervised semantic segmentation model that **strikes a balance between self-supervised and fully supervised paradigms** for visual perception applications in an autonomous driving environment.

In this direction, the thesis is structured into three distinct phases, beginning with self-supervised image classification and progressing toward bi-level image segmentation, ultimately culminating in the development of an advanced semantic segmentation model. Initially, this research employs a simple contrastive learning framework (SimCLR) to classify medical images, specifically focusing on monkeypox diagnosis from skin lesion images, while integrating a federated learning (FL) framework to ensure data privacy. Monkeypox classification is a simple binary classification task and the dataset found for this problem, in this thesis, is very manageable on the computational resources that were available at the onset of this research. It paved the way to grasp non-supervised learning basics and explore how they differ from traditional supervised learning methods.

The subsequent phase involves the development of an efficient convolutional neural network (CNN) with an attention mechanism, applied to the bi-level segmentation task of road pavement crack detection. Similar to the Monkeypox classification, this is also a binary classification task, but at pixel-level, i.e., it is a two-way semantic segmentation problem. Hence, the number of samples found in the relevant datasets is once again manageable on the computational resources available during the research. That formed the basis for learning the basics of deep learning (DL)-

based image segmentation and establishing a solid foundation for the main objective of building an advanced semantic segmentation model. Note that the segmentation model built to solve this problem is the architecture that is reused in the generator sub-module of the generative adversarial network (GAN) developed for the semi-supervised semantic segmentation task in the next phase of the thesis.

In the final phase, the research leverages insights from previous stages to construct an enhanced semi-supervised semantic segmentation model. This model incorporates an attention-driven adversarial training strategy within a GAN framework, designed to improve model performance. The proposed method generates realistic segmentation maps for unlabeled data while enhancing the model accuracy on labeled data. A novel patch-wise discriminator is introduced to extract rich contextual information, further boosting model efficacy.

Extensive ablation studies conducted on widely adopted benchmark datasets across all three phases of the research demonstrate the effectiveness of the proposed models, achieving state-of-the-art performance. The findings of this research contribute to the advancement of semi-supervised learning in computer vision, offering a practical approach to improving model performance while reducing reliance on labeled data.

Table of Contents

List of Figures

List of Tables

List of Important Acronyms

Acronym & its Full Form	**Synopsis**
BCE: Binary Cross Entropy	A loss function used in binary classification tasks, where the model predicts one of two possible classes.
CNNs: Convolutional Neural Networks	CNNs are a specialized type of deep learning model designed for processing grid-like data, such as images. They use convolutional layers to apply filters that capture spatial hierarchies and patterns in the data.
DNN: Deep Neural Network	A type of machine learning model consists of multiple layers of interconnected neurons (deep network) that process and learn from sample data points.
DL: Deep Learning	A subset of machine learning that focuses on using artificial neural networks with many layers to automatically learn and extract features from large amounts of data.
DeepLab	A deep learning architecture designed for semantic image segmentation. It incorporates atrous (or dilated) convolutions to capture multi-scale contextual information and uses a fully connected Conditional Random Field (CRF) to refine the results. There are several versions (e.g., DeepLabv3, DeepLabv3+), each introducing improvements in handling segmentation tasks with greater object delineation.

Acronym & its Full Form	**Synopsis**
DANet	A CNN that uses Dual Attention Network (DAN) incorporating two types of attention mechanisms: spatial attention and channel attention to image segmentation.
DSSL: Deep semi-supervised learning	An advanced technique that leverages both labeled and unlabeled data for training deep learning models, such as DNNs or CNNs.
FL: Federated Learning	A distributed machine learning approach where multiple devices (clients) collaboratively train a shared model while keeping their data local. This method enhances data privacy by ensuring that raw data remains on the clients' devices, and only model updates are shared and aggregated.
FCNs: Fully Convolutional Networks	FCNs are structured entirely using convolutional layers, without relying on densely connected layers like in DNNs at the top. This design allows FCNs to produce output maps with spatial dimensions, making them particularly well-suited for tasks, such as semantic segmentation.
Faster R-CNN: Faster Region-Based Convolutional Neural Network	An advanced object detection CNN that incorporates a Region Proposal Network (RPN) to generate region proposals more efficiently to detect objects accurately.
FLOP: Floating Point Operation	A measure of the computational complexity and efficiency of a computing model, particularly in terms of its ability to handle floating-point operations.
GFLOP: Giga Floating Point Operations Per Second	A unit of measurement used to indicate the performance of a computing model, particularly in terms of its ability to handle floating-point operations.

Acronym & its Full Form	**Synopsis**
GANs: Generative Adversarial Networks	A class of machine learning models, where two networks– a generator and a discriminator–are trained simultaneously in a competitive process. The generator creates fake data, while the discriminator distinguishes between real and fake data, leading to the generation of increasingly realistic outputs
IoU: Intersection Over Union	A metric used to measure the overlap between the predicted regions (segmentation map or bounding box) and the ground truths.
IoMT: Internet of Medical Things	A network of connected medical devices, software applications, and health systems that communicate and exchange data over the internet for monitoring, diagnosing, and treating patients remotely, improving healthcare efficiency and patient outcomes.
LeakyReLU: Leaky Rectified Linear Unit	A type of activation function used in neural networks to address the limitations of the standard Rectified Linear Unit (ReLU) activation function that outputs zero for negative inputs, which can lead to inactive neurons during training.
ML: Machine Learning	A subset of artificial intelligence (AI) that can learn from collected data samples to perform a specific task.
Mask R-CNN: Mask Region-Based Convolutional Neural Network	An extension of the Faster R-CNN model. It not only detects objects in an image and generates bounding boxes, but also adds a branch to predict a pixel-level segmentation mask for each detected object, making it suitable for tasks requiring both object detection and segmentation.

Acronym & its Full Form	**Synopsis**
PCA: Principal Component Analysis	A statistical technique that performs sub-space transformation of input by finding a set of linearly uncorrelated variables called principal components while retaining as much variance as possible. It is mainly used to reduce the dimensionality of datasets or input features of an ML model.
PSPNet: Pyramid Scene Parsing Network	A deep learning model developed for semantic segmentation, which captures both local and global context in an image by employing a pyramid pooling module (i.e., a multi-level feature learning) to better understand spatial hierarchies, improving segmentation accuracy in complex scenes.
ReLU: Rectified Linear Unit	A non-linear activation function commonly used in neural networks.
SiLU: Sigmoid Linear Unit	An activation function that combines both linear and non-linear characteristics.
SimCLR: Simple Contrastive Learning of Representations	A framework for self-supervised learning. It learns visual representations by maximizing the agreement between augmented views of the same image and minimizing the agreement between different images. It utilizes an advanced loss function–contrastive loss and relies on data augmentation and a large encoder model to learn useful feature representations without requiring labeled data.
SSL: Semi-Supervised Learning	A deep learning model cable of learning hidden patterns and structures entirely from unlabeled data without explicit ground truths.
Tanh: Hyperbolic Tangent	A zero-centered activation function in neural networks.

Acronym & its Full Form	**Synopsis**
MAE: Mean Absolute Error	A metric used to measure the average magnitude of errors in a set of predictions.
U-Net	A type of CNN particularly developed for image segmentation tasks. It has a U-shaped architecture, with an encoder subnetwork that captures context and a symmetric decoder subnetwork that enables precise localization, making it effective for pixel-wise segmentation.
MAE: Mean Absolute Error	A metric used to measure the average magnitude of errors in a set of predictions.
MSE: Mean Squared Error	A common metric used to measure the average squared difference between predicted and actual values.
mIoU: Mean Intersection Over Union	A commonly used metric for evaluating the performance of segmentation models, particularly in semantic segmentation tasks, across multiple samples.

Chapter 1

Introduction

1.1 Thesis Overview

Computer vision, the field of computing that allows computers to interpret and comprehend the visual environment, has advanced dramatically with the introduction of deep learning. Deep learning (DL), particularly convolutional neural networks, has transformed several computer vision applications, including image classification, object detection, and image segmentation. Recently, in the realm of computer vision, semi-supervised learning (SSL) has emerged as a pivotal approach for leveraging both labeled and unlabeled data, addressing the limitations of supervised learning approaches that require large amounts of labeled datasets. Despite the advancement in SSL, there are still several challenges in this field. Key issues include the effective integration of labeled and unlabeled data, the risk of propagating errors from incorrect pseudo-labels, and the need for robust algorithms that can generalize well across various tasks. Moreover, designing SSL methods that efficiently utilize computational resources while maintaining high accuracy remains a significant challenge. The complexity of these problems is heightened when transitioning from simpler tasks like classification to more complex ones like image segmentation, which require detailed pixel-level predictions.

This thesis leverages cutting-edge advancements in DL and SSL to develop improved models for semantic segmentation—paving the way for transformative applications in computer vision.

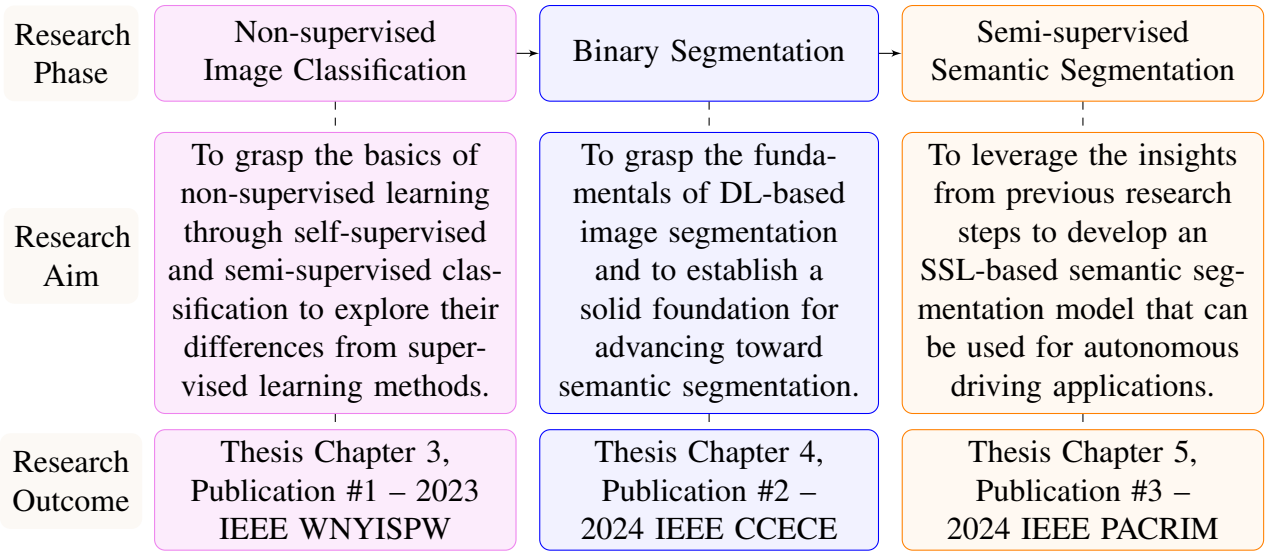


Figure 1.1: Illustration of the thesis roadmap. It subsumes three stages, each dedicated to acquiring essential knowledge needed to achieve the final objective of the thesis.

As illustrated in Fig. 1.1 this thesis adopts a pragmatic approach, systematically building a robust knowledge base before advancing to the development of an improved semi-supervised semantic segmentation model, which is the central objective of the thesis. Given the complexity of semantic segmentation in computer vision, the study begins with an exploration of a self-supervised image classification model to grasp the fundamental concepts of various machine learning approaches. It is worth noting that if a self-supervised model is fine-tuned on a small amount of labeled data after being initially trained on unlabeled data, it can effectively function as a semi-supervised model. Thus, the first research phase lays the groundwork for understanding general non-supervised learning approaches and distinguishing them from traditional supervised methods (cf. Chapter 3).

In the second phase, a binary segmentation model is constructed to solidify the principles of DL-based image segmentation, providing a strong foundation for progressing toward more intricate semantic segmentation tasks (cf. Chapter 4). Finally, in the last phase by leveraging the insights gained from the previous two phases, this thesis culminates in the development of an improved semi-supervised learning framework for image semantic segmentation, specifically tailored for autonomous driving applications (cf. Chapter 5), thereby achieving the main objective.

1.2 Motivation

Traditional supervised deep learning approaches require large amounts of labeled data to achieve high performance. However, acquiring such data often involves human expertise, domain knowledge, and time, making the data acquisition process expensive. In some cases, privacy concerns make collecting such samples impractical. For example, in medical imaging, annotating each pixel in a scan to distinguish between tumors and healthy tissue requires substantial time and expertise from medical professionals. Similarly, in satellite imagery, identifying different land covers or tracking changes over time necessitates detailed analysis by experts. In autonomous driving, labeling road cracks or segmenting different objects in the scene (e.g., roads, pedestrians, and vehicles) is labor-intensive and costly, as it requires pixel-level annotations across multiple semantic objects.

On the other hand, SSL offers a solution by leveraging a smaller set of labeled data alongside a larger collection of unlabeled data, thereby reducing the dependency on extensive labeled data samples to build a deep learning model for a given application. SSL helps models generalize better by incorporating unlabeled data, particularly in scenarios with limited labeled data, leading to improved accuracy and robustness as the models are exposed to a wider range of examples. In medical imaging, for instance, SSL can use vast amounts of available but unlabeled scans, such as Magnetic Resonance Imaging (MRI), to enhance the model's understanding of normal and pathological variations, thereby improving diagnostic accuracy. In satellite image analysis, SSL can utilize extensive archives of unlabeled images to improve the model's ability to detect and classify various land covers and changes, enhancing environmental monitoring and resource management. For autonomous driving, SSL can employ large volumes of unlabeled driving footage to better identify road cracks and segment different elements of the driving environment, thus enhancing vehicle safety and navigation.

Moreover, with the growing availability of unlabeled data, SSL provides a scalable solution for training models without requiring a proportional increase in labeled data. This efficiency is crucial as datasets expand in size and complexity. For example, as new medical imaging technologies produce more detailed and higher-resolution scans, the amount of available unlabeled data increases

exponentially. Similarly, as satellite imagery continuously captures the Earth's surface, the volume of unlabeled data grows rapidly. In autonomous driving, the constant stream of driving data from sensors and cameras leads to an ever-growing pool of unlabeled data. SSL can effectively harness this data, enabling the development of more accurate and robust models without a corresponding increase in labeling efforts.

In conclusion, SSL for image classification and segmentation provides a promising approach to overcoming the challenges associated with acquiring labeled data. By effectively leveraging unlabeled data, SSL can enhance model performance, reduce reliance on costly manual annotations, and offer scalable solutions for managing large and complex datasets. This makes SSL a crucial technique for advancing fields such as medical imaging, satellite image analysis, and autonomous driving. However, The development of a model using SSL involves the following key challenges: effective use of unlabeled data, handling class imbalance and bias in labeled data, inaccuracies in pseudo-labeling, determining the optimal labeled data proportion, selection of data augmentation and regularization functions, computational complexity, and interpretability and explainability.

1.3 Technical approach

In this study, the focus is on developing advanced semi-supervised learning models aimed at enhancing image classification and segmentation. The approach is structured into three main stages: feature extractions, model building and training, and evaluation. This research introduces a robust semi-supervised approach that leverages both labeled and unlabeled datasets to achieve accurate image semantic segmentation. These solutions are derived from extensive experimental studies conducted on well-known benchmark datasets. By combining limited labeled data with a larger pool of unlabeled data, the proposed models are intended to perform the desired tasks more effectively. Finally, a thorough comparative analysis of the proposed models' performance against state-of-the-art methods will be conducted, highlighting their superiority and efficiency.

1.4 Overview of Computer Vision: Concepts and Applications

Computer vision is a branch of artificial intelligence that aims to enable computers to process and understand visual information from the world, similar to how humans perceive their surroundings. Computer vision encompasses several research domains, each of those domains addressing unique challenges and driving the overall advancement of the field. Fig. 1.4 illustrates the practical applications of computer vision research. However, this study focuses on two core tasks: image classification and image segmentation.

1.4.1 Image Classification

Image classification is the process of assigning a label or category to a complete image based on its contents. For instance, in a dataset of animal photos, the objective is to identify whether an image depicts a cat, dog, bird, or another animal. For recent advancements in deep learning, especially with convolutional neural networks, the accuracy and efficiency of image classification have seen remarkable improvements. These networks can learn to identify complex features within images, allowing them to accurately distinguish between different classes. Fig. 1.2 shows an example of a binary classification task performed by a simple neural network.

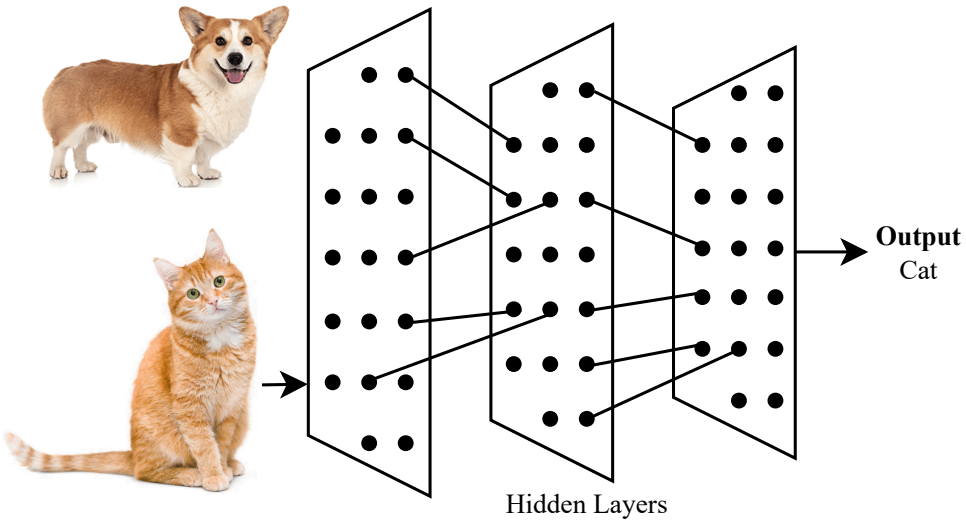


Figure 1.2: An illustration of deep learning-based image classification using a three-hidden layer model that performs cat vs dog classification.

In image classification, usually, the process begins with preprocessing steps such as resizing, normalization, and data augmentation to prepare the images for the neural network. CNNs then use multiple layers of filters to extract hierarchical features from the images, starting from low-level features like edges and textures to high-level features representing specific objects or patterns. The extracted features are then fed into fully connected layers, culminating in a classification layer that calculates the probability distribution over the predefined classes.

Recent research has introduced advanced architectures, like the Residual Networks (ResNet) [4], Densely Connected Convolutional Networks (DenseNet) [5], and Efficiency-focused Networks (EfficientNet) [6], which have pushed the boundaries of deep learning models for image classification and related applications. Meanwhile, transfer learning has also become a popular approach, where pre-trained models on large datasets, like ImageNet are fine-tuned on selective tasks, significantly reducing the training time and improving performance.

1.4.2 Image Segmentation

Image segmentation is a more intricate and detailed task than image classification. It is the process of labeling the target image into regions, whereby pixel-level labels are generated. It has been a cornerstone for several computer vision-based applications since pixel-level classification captures intricate details of the input.

However, per-pixel annotations are not readily scalable, especially for large-scale datasets and complex tasks, due to the following reasons: (i) Labeling each pixel of an image requires a significant amount of manual effort and time, which becomes impractical for extensive datasets; (ii) Accurate per-pixel labeling demands domain expertise and meticulous attention to detail, making it challenging to ensure consistency and high quality; (iii) The volume of data in semantic segmentation tasks is often large, leading to high computational and storage requirements for supervised learning; (iv) Maintaining consistency and precision across all pixel labels is difficult, and errors in labeling can negatively impact model performance. To address these issues, Semi-Supervised and Self-Supervised Learning approaches are increasingly utilized to reduce reliance on large amounts of labeled data by leveraging unlabeled data and pre-trained models.

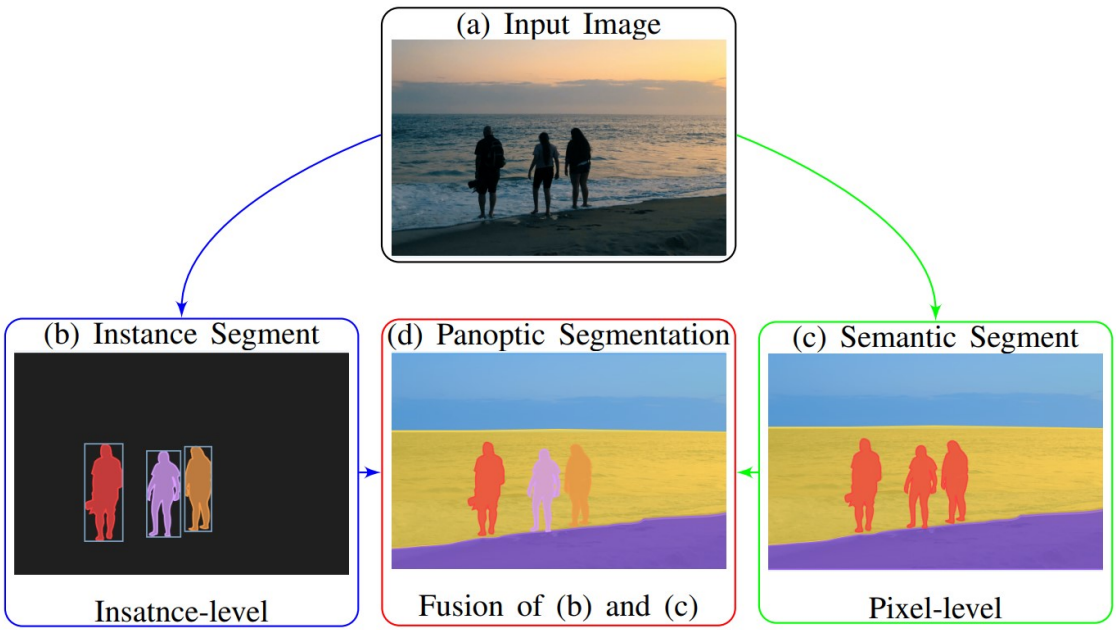


Figure 1.3: Types of image segmentation: (a) an input, (b) instance segmentation with per-object mask and class label, (c) semantic segmentation with per-pixel class labels, and (d) panoptic segmentation with per-pixel class and instance-level labels.

Referring to Fig. 1.3, image segmentation can be broadly categorized into instance segmentation that detects and segments individual objects within an image, distinguishing between different instances of the same class [7–10], semantic segmentation that classifies each pixel in an image into predefined categories, providing a detailed understanding of the scene's structure and content [11, 12], and panoptic segmentation [13–16], which combines the first two approaches to detect and segment each object instance while simultaneously classifying each pixel into a semantic category, providing a comprehensive understanding of both object identities and their spatial layout. Semantic segmentation, in particular, has garnered significant interest across sectors, such as agriculture, healthcare, transportation, and infrastructure management [17].

Various techniques can perform image segmentation, including edge-based, cluster-based, and region-based methods. These methods can be implemented using supervised (learning from only labeled data to map inputs to correct outputs), unsupervised (learning patterns and structures entirely from unlabeled data without explicit ground truths), semi-supervised (leveraging a small amount of labeled data along with a large amount of unlabeled data), or self-supervised (gen-

erating labels from the data itself, learning useful representations by solving pretext tasks without needing manually labeled data) machine learning approaches [18, 19] as elaborated in Section 1.5.1. For medical image segmentation, CNNs are a widely used backbone model [20]. If CNNs are considered for such medical image segmentation, the following three major challenges should be overcome: (i) handling issues of high-resolution inputs, (ii) handling multiple objects with various scales, and (iii) maintaining high object localization accuracy [21]. In this direction, Chaudhary *et al.* [22] reported a semi-supervised model for eye image segmentation, in which they utilized a domain-specific augmentation to perform a pretext task.

On the other hand, self-supervised learning approaches are also used for medical image analysis. The model developed by Dosovitskiy *et al.* [23] is one of the earliest such methods that integrate a discriminative unsupervised feature learning scheme with a CNN, called Exemplar-Net. Here, the model learns from unlabeled raw data, without needing manually annotated ground truths. Different pretext tasks were used for pre-training, and then the model will be retrained to perform a downstream task using a few labeled data. However, the author used self-supervised learning for segmentation without fine-tuning. It is found in [24] that the model boosts the segmentation performance after fine-tuning with some labeled data. In addition, the existing self-supervised learning approaches still use manual annotation for fine-tuning the model with some labeled data. This is because the self-supervised approach follows two steps. In the first step, the model learns from representation learning using different pretext tasks, like augmentation, and colorization. After that, it performs a downstream task, i.e., the pre-trained model is tested for a task-specific operation using supervised or unsupervised learning approaches.

Similarly, image segmentation is also necessary for the agricultural sector, where it is predominantly used for insect detection, leaf disease segmentation, flower segmentation, and crop segmentation. For instance, Wang *et al.* [25] proposed a conventional ML approach using k-means clustering to segment types of crop insect pests. Similarly, Guldenring and Nalpantidis [26] worked on agricultural image perception using SSL. They used a contrastive learning approach to perform required pretext tasks. After that, they performed downstream tasks, including classification and segmentation.

In summary, it is found that semi-supervised learning approaches for image segmentation have gained attention in computer vision applications across various fields, from medical imaging and autonomous driving to satellite imagery, scene perception, and industrial inspection [22, 27–32]. These approaches leverage large amounts of unlabeled data along with a smaller set of labeled data to enhance the performance and scalability of segmentation models, making them more robust and generalizable to diverse tasks.. These approaches leverage large amounts of unlabeled data along with a smaller set of labeled data to enhance the performance and scalability of segmentation models, making them more robust and generalizable to diverse tasks.

1.4.3 Applications of Computer Vision

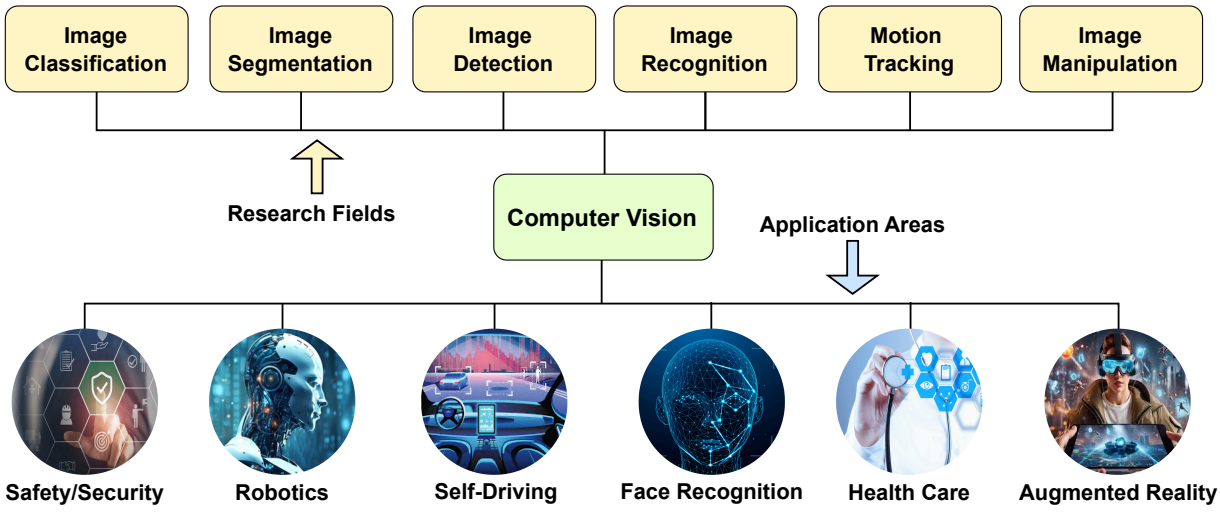


Figure 1.4: Overview of computer vision research fields and their applications.

Referring to Fig. 1.4, in safety and security, computer vision technologies are used for surveillance, threat detection, and monitoring, enhancing public safety. In robotics, these technologies enable robots to interact more effectively with their environment, facilitating navigation and object manipulation. Self-driving cars rely heavily on computer vision for real-time perception and decision-making, using techniques such as image segmentation and detection to navigate and avoid obstacles. Health care is another significant application area, where computer vision is transforming medical diagnostics and patient care. Techniques like image classification and segmentation

are used to analyze medical images, aiding in disease detection and diagnosis. Face recognition technology, under image recognition, is widely used for identity verification and access control, enhancing security. Augmented reality leverages computer vision to overlay digital information in the real world, enriching user experiences in gaming, education, and retail.

In the context of semi-supervised learning, it's possible to improve the generalization and robustness of a model by exposing it to a diverse set of examples during training. In medical imaging, for instance, SSL can utilize vast amounts of unlabeled scans to improve diagnostic accuracy. In autonomous driving, SSL can better identify and segment road features and obstacles by leveraging the continuous stream of unlabeled driving footage. Overall, SSL in computer vision reduces the dependency on extensive manual labeling, making it a cost-effective and efficient approach to handling large and complex datasets.

1.5 Deep Learning

1.5.1 The Basic Deep Learning Paradigms

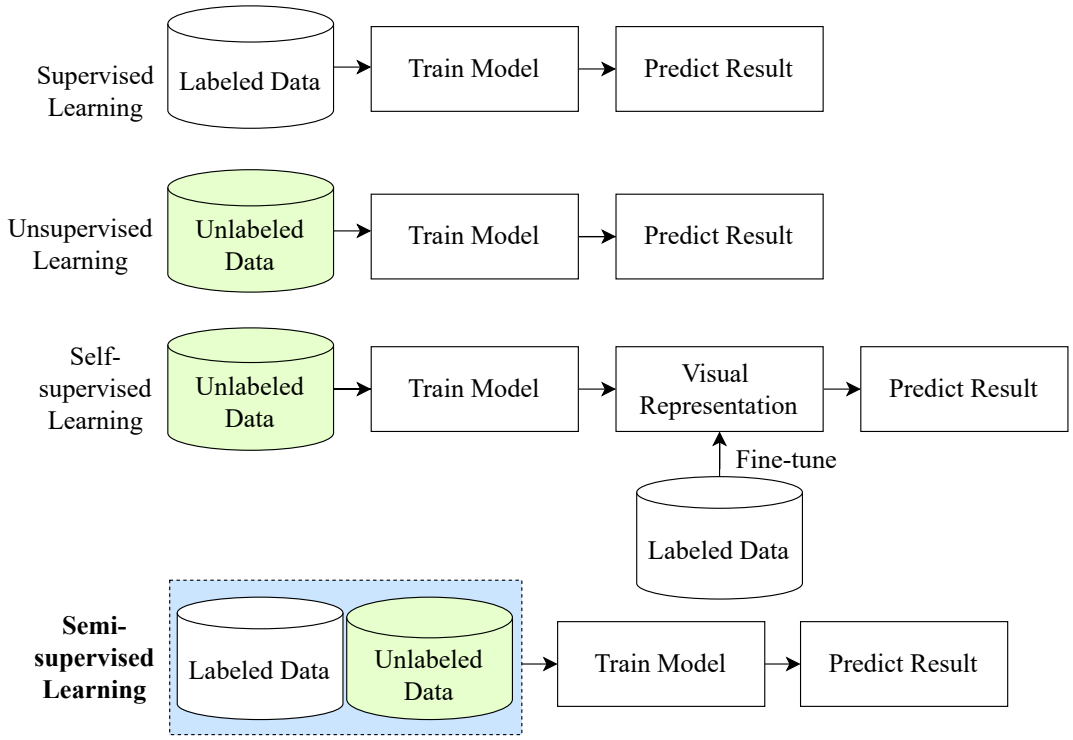


Figure 1.5: An overview illustrating the basic concepts of different machine learning paradigms.

As illustrated in Fig. 1.5, deep learning has experienced tremendous progress and diversified into supervised, unsupervised, self-supervised, and semi-supervised approaches based on their type of learning pipeline.

Supervised Learning

The supervised learning strategies rely heavily on large volumes of well-annotated data [33]. Unfortunately, obtaining such labeled data can be challenging, costly, and time-intensive. On the other hand, unlabeled data is often plentiful and more accessible or affordable to collect. Therefore, it becomes advantageous to harness this vast pool of unlabeled data to enhance learning performance when only a limited number of labeled samples are available. This need has made semi-supervised learning a significant focus in machine learning research over the past decade [34]. For readers

interested in exploring SSL further, the following source is recommended - Deep Learning, An MIT Press book, by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.

Unsupervised Learning

Unlike supervised learning, unsupervised learning involves training a model on data without labeled responses or outputs, to discover hidden patterns, structures, or relationships within the data. Common techniques include clustering, where the model groups similar data points together, and dimensionality reduction, which reduces the number of features while preserving essential information. Examples of unsupervised learning algorithms include k-means clustering, hierarchical clustering, and Principal Component Analysis (PCA). Unsupervised learning is beneficial for tasks such as anomaly detection, data exploration, and feature extraction, where understanding the data's structure is the primary goal without the need for predefined labels. For readers interested in exploring SSL further, the following source is recommended - Unsupervised Learning: Foundations of Neural Computation, Edited by Geoffrey Hinton, and Terrence J. Sejnowski.

Self-supervised Learning

Self-supervised learning is a machine learning paradigm where the model learns directly from the data without relying on explicit labels. Instead, the model generates its own pseudo labels through pretext tasks, which are specifically designed to help the model uncover useful data representations. For instance, in image data, a self-supervised model might be trained to predict missing parts of an image or to infer the context surrounding a specific patch. The features learned during this process can later be fine-tuned or transferred to supervised tasks, such as classification or object detection. By leveraging the inherent structure and patterns within the data, self-supervised learning is particularly valuable in scenarios where labeled data is limited. For readers interested in exploring SSL further, the following source is recommended - A Cookbook of Self-Supervised Learning.

Semi-supervised Learning

Since its introduction in the 1970s, semi-supervised learning has given rise to a wide array of methodologies, such as generative models, semi-supervised support vector machines, graph-based methods, and co-training approaches [35]. In recent years, deep neural networks have gained prominence in many research areas, and there is a growing need to adapt the classic SSL frameworks to these settings. This adaptation has led to the development of deep semi-supervised learning (DSSL), which explores how deep neural networks can effectively utilize both labeled and unlabeled data. Numerous DSSL methods have been proposed and applied across various tasks and domains, including image classification, object detection, semantic segmentation, text classification, and sequence learning.

SSL sits between supervised and unsupervised learning, leveraging both labeled and unlabeled data to boost model accuracy. This approach is especially useful in situations where acquiring labeled data is challenging or costly. By reducing the need for extensive labeled datasets, SSL has become a valuable method in various fields. In healthcare, SSL proves highly beneficial, particularly in medical imaging, where obtaining annotated data is both expensive and time-consuming. For example, radiologists manually label scans to detect conditions such as tumors or fractures, a process requiring specialized knowledge and significant effort. To tackle this, SSL uses a small set of labeled medical images along with a larger pool of unlabeled images.

On the other hand, in the realm of self-driving cars, SSL is crucial for refining vehicle perception systems, which handle tasks like object detection, lane detection, and semantic segmentation. These capabilities are essential for safe navigation, helping the vehicle understand its environment, recognize obstacles, and make informed driving decisions. Given that labeled data for rare or hazardous driving scenarios is often limited, SSL becomes particularly valuable. For example, a semi-supervised model trained to recognize pedestrians or cyclists can start with a small set of labeled images and expand its learning using a vast amount of unlabeled driving footage. Techniques such as adversarial training, where one model generates pseudo-labeled images for another to learn from, and co-training, where multiple models learn from each other's predictions, are used to enhance the model's performance.

The primary strength of SSL is its capacity to utilize large amounts of readily available unlabeled data, thus reducing reliance on extensive labeled datasets. While SSL can improve model generalization and performance, its effectiveness depends on the quality of the pseudo-labels and the model's ability to manage noise and inaccuracies. Future advancements in SSL will likely focus on enhancing the reliability of pseudo-labels and developing computationally efficient methods, making SSL an even more powerful tool in various applications. In general, generative adversarial networks (GANs) are exploited to build the semi-supervised learning process. For readers interested in exploring SSL further, the following source is recommended - the SSL book.

1.5.2 Generative Adversarial Networks

The GANs, consisting of a discriminator and a generator subnetworks, have been widely used for developing semi-supervised learning models. In the context of semi-supervised learning, the discriminator is often modified to perform two tasks simultaneously: distinguishing between real and generated data and classifying real data into its respective categories. This dual role allows the discriminator to benefit from the generated data, even though it's not explicitly labeled, thereby improving its ability to generalize from the labeled data. A notable advantage of using GANs in semi-supervised learning is their capability to enhance performance with limited labeled datasets, which is especially valuable in fields like medical imaging, where acquiring labeled data is costly and time-consuming [36]. A seminal work in this area demonstrated that adding a small number of labeled samples to a GAN could drastically improve classification performance compared to purely unsupervised or fully supervised methods with the same amount of labeled data [37]. This approach has inspired various extensions and improvements, further solidifying the role of GANs as a critical tool for semi-supervised learning in tasks that range from image recognition to natural language processing. For readers interested in exploring GNNs further, the following source is recommended - the GAN book.

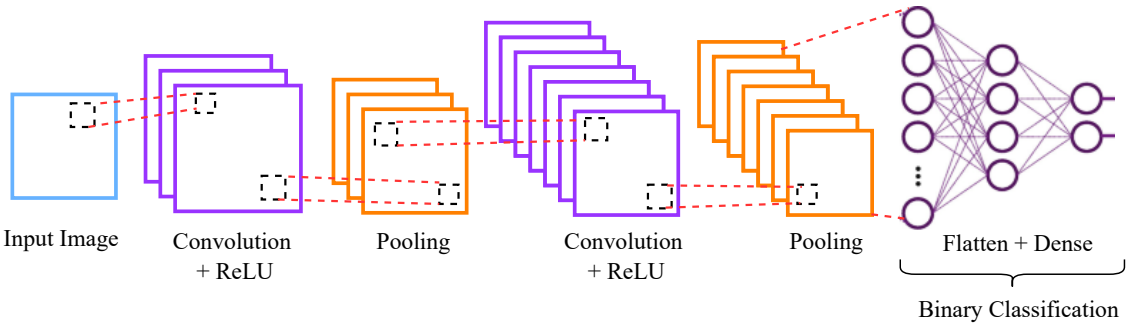


Figure 1.6: Overview of a basic two-way classification CNN. It denotes the key layers, like the convolution, pooling, and dense layers, and the function–ReLU placed for non-linear activation.

1.5.3 The Advancements in Deep Learning Models for Image Classification

Deep learning for image classification and segmentation are advanced neural network models designed to analyze and understand the visual representation of data with high accuracy. For image perception, CNNs are widely used. A vanilla (i.e., basic) CNN, as shown in Fig. 1.6 consists of several convolutional layers that detect complex patterns and features with the help of convolution operations, non-linear activation functions (e.g., ReLU, SiLU), pooling layers that reduce dimensionality while preserving essential information, and fully connected layers that aggregate these features to produce a classification label for the input image as a whole.

In contrast, image segmentation aims to label each pixel of an image according to its class, providing a detailed segmentation map. Architectures like the U-Net in [38], are specifically designed for this purpose, employing an encoder-decoder structure with skip connections to retain spatial information and produce precise segmentation results. The Fully Convolutional Networks also play a crucial role by applying convolutional operations to the entire image, enabling pixel-wise predictions. These deep learning models are instrumental in achieving accurate and detailed analysis of images, leveraging complex patterns and relationships learned from extensive training data.

Over the past two decades, deep learning has revolutionized several computer vision tasks with the emergence of CNNs. The following subsections highlight a few milestone models from the literature.

AlexNet

AlexNet [39] was one of the pioneering CNN architectures that demonstrated the potential of deep learning for image classification. It consists of five convolutional layers followed by three fully connected layers. It uses the ReLU (cf. Section 1.5.5) activation function and incorporates dropout layers to prevent overfitting. Hence, the application of data augmentation techniques significantly contributed to its success in the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC).

GoogLeNet (Inception)

GoogLeNet [40], also known as Inception, introduced the concept of inception modules, which allow the network to capture multi-scale information by performing convolutions with different kernel sizes within the same layer. This architecture drastically reduced the number of parameters compared to earlier CNNs while maintaining high accuracy.

ResNet

Residual Networks (ResNet) [4] introduced residual connections, also known as skip connections, to address the issue of vanishing gradients in particularly deep networks. Gradients can flow straight across these connections, which makes it possible to train incredibly deep networks. In several image classification benchmarks, ResNet designs, including ResNet-50 and ResNet-101, have demonstrated state-of-the-art performance.

DenseNet

Densely Connected Networks (DenseNet) [5] further improved the flow of information and gradients by connecting each layer to every other layer in a feed-forward fashion. This dense connectivity pattern leads to improved feature reuse and reduced vanishing gradient problems, resulting in highly efficient and accurate models.

EfficientNet

With the help of a straightforward yet efficient scaling coefficient, EfficientNet [6] developed a compound scaling technique that uniformly scales network depth, width, and resolution. With this method, a family of models that outperform earlier architectures in terms of efficiency and accuracy can be produced.

1.5.4 The Advancements in Deep Learning Models for Image Segmentation

Classifying each pixel in an image into specified categories is the key aspect of semantic segmentation, which is more complicated than image classification. The following deep learning architectures have significantly advanced this field.

Fully Convolutional Networks (FCNs)

FCNs [41] were the first deep learning models to replace fully connected layers with convolutional layers, enabling pixel-wise prediction for semantic segmentation. By using deconvolutional layers (also known as transposed convolutions) to upsample the feature maps, FCNs produce dense predictions that match the input image resolution.

U-Net

U-Net [38] is an encoder-decoder architecture with symmetric skip connections that link corresponding layers in the encoder and decoder paths shown in Fig. 1.7. These skip connections help retain spatial information lost during downsampling, making U-Net highly effective for biomedical image segmentation and other applications requiring fine-grained segmentation. Overall, U-Net has become one of the most popular architectures for image segmentation due to its powerful and efficient design.

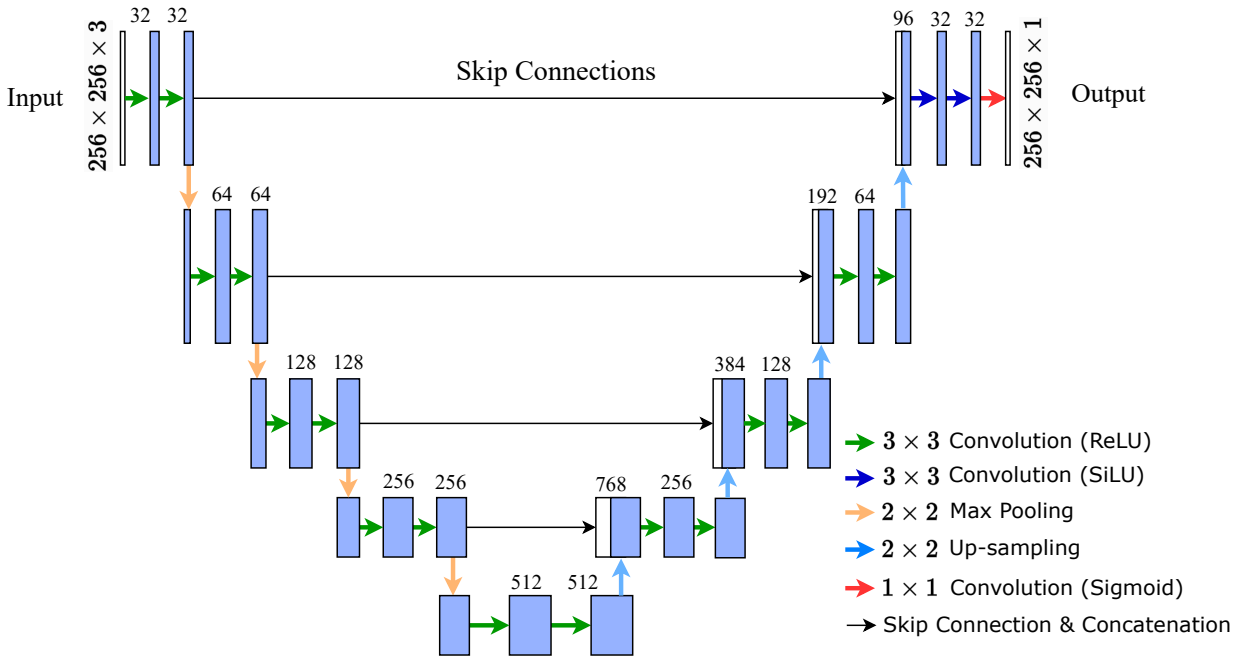


Figure 1.7: A basic architecture of the U-Net-based image segmentation.

SegNet

SegNet [42] is another encoder-decoder architecture designed for semantic segmentation, similar to the U-Net. It uses a series of convolutional and pooling layers for encoding, followed by upsampling layers using max unpooling layers that utilize the pooling indices from the encoder to ensure that the spatial information is preserved, resulting in improved segmentation results and computational efficiency. In SegNet, only the pooling indices are transferred from the encoder path to the decoder path, requiring less memory. In contrast, U-Net transfers entire feature maps, requiring more memory.

DeepLab

DeepLab [43] introduced atrous (dilated) convolutions to capture multi-scale context by expanding the receptive field without increasing the number of parameters. Various versions of DeepLab, such as DeepLabv3 and DeepLabv3+, have integrated atrous spatial pyramid pooling (ASPP) and encoder-decoder structures to enhance segmentation performance.

Mask R-CNN

Although primarily known for instance segmentation, Mask R-CNN [44] can also be used for semantic segmentation. It extends Faster R-CNN by adding a branch for predicting segmentation masks on each Region of Interest (RoI), providing precise pixel-level object detection and segmentation.

HRNet

High-Resolution Network (HRNet) [45] maintains high-resolution representations throughout the network by connecting high-to-low resolution convolutions in parallel. This approach allows the model to capture detailed spatial information and achieve high accuracy in semantic segmentation tasks.

PSPNet

Pyramid Scene Parsing Network (PSPNet) [46] employs a pyramid pooling module to gather contextual information at multiple scales. By aggregating global context features, PSPNet improves the model's ability to recognize objects of varying sizes and enhances overall segmentation accuracy.

Deep learning models have significantly advanced the fields of image classification and semantic segmentation. Architectures like AlexNet, GoogLeNet, ResNet, DenseNet, and EfficientNet have set new benchmarks in image classification, while FCNs, U-Net, SegNet, DeepLab, Mask R-CNN, HRNet, and PSPNet have revolutionized semantic segmentation. These advancements have enabled a wide range of applications, from autonomous driving and medical imaging to video surveillance and augmented reality, demonstrating the transformative potential of deep learning in computer vision.

1.5.5 Common Activation Functions in Deep Learning

In deep learning, activation functions are crucial for capturing the non-linear relationships between inputs and outputs, significantly enhancing the network's ability to model complex patterns. Various activation functions have been developed, each offering unique advantages. Fig. 1.8 visualizes the behavior of the most common activation functions used in modern deep neural networks.

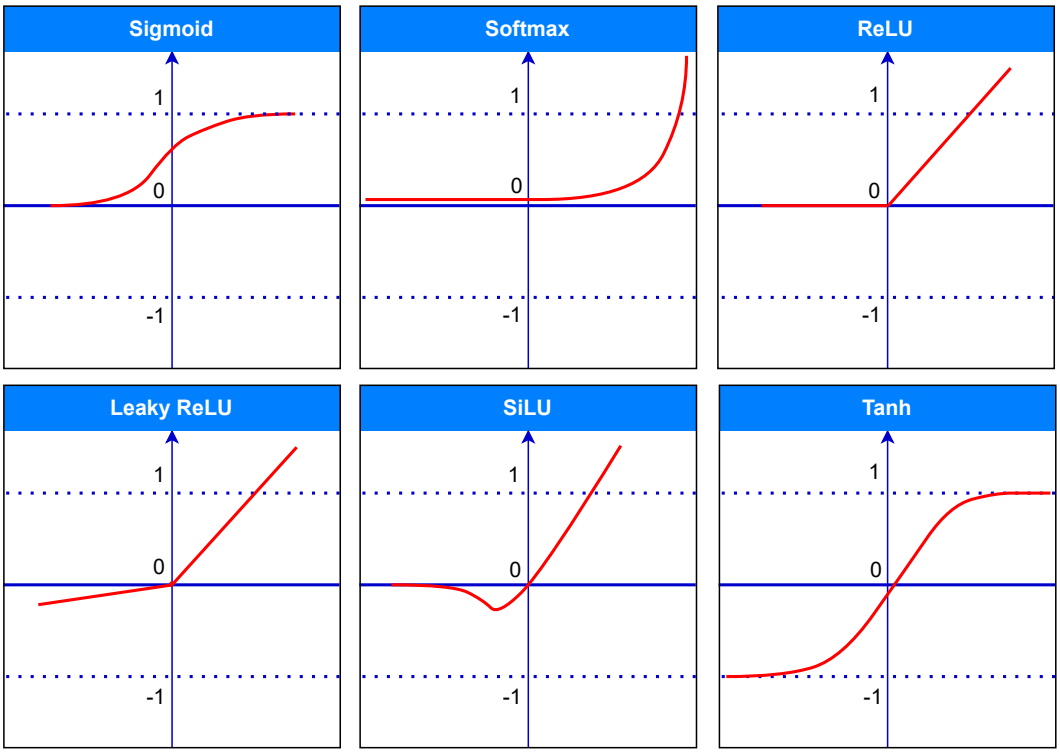


Figure 1.8: Visualization of the characteristics of common activation functions used in DL.

Choosing the right activation function is often an empirical process, involving experimentation to find the most suitable function for a specific application. The following subsections elaborate on the activation functions depicted in Fig. 1.8.

Sigmoid

The sigmoid activation function maps any input to a value between 0 and 1, making it useful for binary classification problems. It is mathematically defined as $\sigma(\mathbf{x})$ in (1.1).

$$\sigma(\mathbf{x}) = \frac{1}{1 + e^{-x}}, \tag{1.1}$$

where x is the input.

Softmax

The softmax function is commonly applied in the output layer of a neural network for multiclass classification. It transforms the outputs from multiple neurons into a probability distribution across various classes. The softmax function, $S(\mathbf{z_i})$ is defined in (1.2).

$$S(\mathbf{z_i}) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}, \tag{1.2}$$

where z_i denotes the output of the i-th neuron, K is total number of classes, and j spans all neurons in the output layer. This ensures that the sum of probabilities equals 1.

Rectified Linear Unit also known as ReLU

It is extensively employed in deep learning due to its simplicity and computational efficiency. It operates by zeroing out negative input values while preserving positive values unchanged, thereby introducing non-linearity into the model. This activation function plays a critical role in addressing the vanishing gradient problem, facilitating more effective learning and convergence in deep neural networks. The ReLU, $F(\mathbf{x})$ is defined as in (1.3).

$$F(\mathbf{x}) = \max(0, x), \tag{1.3}$$

where x is the input. ReLU helps mitigate the vanishing gradient problem. Note that deep learning models often encounter vanishing gradients during training using gradient-based backpropagation

techniques, i.e., optimization algorithms. This occurs when the magnitude of gradients becomes extremely small, hindering the model's ability to learn effectively.

Sigmoid Linear Unit or SiLU

It is also known as the Swish activation function, $\mathrm{Swish}(\mathbf{x})$ is defined as in (1.4).

$$Swish(\mathbf{x}) = x \cdot \sigma(x) = \frac{x}{1 + e^{-x}}, \tag{1.4}$$

where $\sigma(x)$ is the sigmoid function defined earlier in (1.1). SiLU has been shown to improve performance in deep networks by combining the benefits of both linear and nonlinear activations. On the other hand, the SiLU function effectively gates its input by multiplying it with its sigmoid-transformed value. This self-gating mechanism allows the network to retain a certain level of negative input, potentially leading to richer representations and improved learning dynamics, especially in deeper networks.

Hyperbolic Tangent also known as Tanh

The tanh function maps inputs to a range between -1 and 1, providing a zero-centered output, which can be beneficial for optimization. It is given by $\tanh(\mathbf{x})$ in (1.5).

$$\tanh(\mathbf{x}) = \frac{e^{\mathbf{x}} - e^{-\mathbf{x}}}{e^{\mathbf{x}} + e^{-\mathbf{x}}} = \frac{1 - e^{-2\mathbf{x}}}{1 + e^{-2\mathbf{x}}}, \tag{1.5}$$

where $\mathbf{x}$ is the input variable, and $e^{\mathbf{x}}$ and $e^{-\mathbf{x}}$ represent the exponential function and its reciprocal, respectively.

Leaky ReLU

It is a variant of ReLU to allow a small, non-zero gradient when the input is negative, addressing the "dying neuron" that often occurs in DNNs with ReLU activation functions, during gradient-based

training. It is characterized by the expression (1.6):

$$F(\mathbf{x}) = \max(a \cdot \mathbf{x}, \mathbf{x}), \tag{1.6}$$

where a is a small positive constant that defines the slope for negative input values, typically set to 0.01. The user can redefine it during the training of a DL model.

1.6 Thesis contribution

The primary contribution of this thesis is the development of a sophisticated framework based on semi-supervised learning for computer vision applications, mainly targeting image semantic segmentation. For image segmentation, the proposed model utilizes the Generative Adversarial Networks framework. It can be seamlessly integrated into existing computer vision workflows, making it both scalable and practical. The research findings of this thesis pave a path to advance deep learning techniques, particularly in settings, where labeled data is scarce. The key contributions of this study are as follows:

- **Comprehensive Exploration of Data Preprocessing Techniques:** This thesis provides an in-depth exploration of data preprocessing methods, including strategies for handling missing labels, normalizing data, and selecting features informed by domain-specific knowledge.
- **Optimizing Framework Performance:** By exploring and fine-tuning the hyperparameters of the GANs framework, the study optimizes the performance of the image segmentation model. For medical image classification a self-supervised and a secure federated learning framework is also integrated to get an understanding of the performances of semi-supervision.
- **Versatility Across Diverse Datasets:** Several benchmark datasets were used for model evaluation to make the model robust and generalized.

- **Comparative Analysis:** The effectiveness of the proposed GANs framework is rigorously evaluated through comparative analyses, highlighting the framework's improvements over existing methods.
- **Enhanced Efficiency and Reduced Error Rate:** The proposed GAN framework significantly improves the efficiency and accuracy of image segmentation tasks, showcasing notable advancements over current state-of-the-art approaches.

Chapter 2

Related Works

This chapter reviews existing research and literature, identifying gaps in the field and explaining how this study addresses them. It covers image classification techniques and semantic segmentation, providing a comprehensive overview of the current state of research and laying the groundwork for the study's contributions.

2.1 Literature Review on Image Classification DL Models

Deep learning models play a vital role in image classification due to their ability to learn and extract complex patterns from large-scale datasets. In medical diagnosis, like Mpox classification, deep learning algorithms, particularly CNN can effectively analyze intricate textural and structural features that are characteristics of a specific disease from their respective medical imaging [47]. For example, Pramanik *et al.* utilize several pre-trained deep CNN image classifiers, viz. Inception-V3, Xception, and DenseNet169 under an ensemble model for Mpox identification from skin lesion images [48]. To get a refined final prediction, they introduce a customized score function to aggregate the complementary cues learned by the individual learners. Regardless of its high complexity, this approach is reported to achieve an average performance of more than 90% in key evaluation metrics, such as accuracy, precision, recall, and F1-score. Ali *et al.* [1] also propose an ensemble model using a majority voting technique to diagnose skin lesion diseases. They combine three

pre-trained classifiers, such as Inception-V3, VGG16, and ResNet50. Their results show that the ResNet50 achieves the best results when the models are evaluated individually. However, more investigation is needed to prove these results, since the dataset used for the experiments was created using a web-scraping method and contained a limited sample set. Similarly, Sitaula and Shahi [49] fine-tune thirteen pre-trained CNNs for Mpox classification. By combining the best-performing models under a majority voting scheme, the framework produces promising performance. Another fusion-based ensemble model is developed by Liu in [50]. This research focuses on a bi-linear pooling model with a combination of two pre-trained models EfficientNet and DenseNet, where the framework uses bi-linear features. On the other hand, to avoid the complexity of an ensemble model, authors employ Mini-GoogLeNet in [51] with a small training dataset to avoid overfitting issues. However, to gain trust and deploy it on real-world applications, it is important to train the models on a diverse and large amount of samples.

To address issues and to create an efficient data-crunching pipeline, self-supervised and semi-supervised learning-based solutions have gained enormous attention. Because these learning approaches can generate useful feature maps from unlabeled samples during the training phase. This property is particularly beneficial in medical image analysis, where expert annotation is costly and time-consuming. Employing self-supervised and semi-supervised learning can significantly reduce the burden of data annotation, making it a practical solution for healthcare applications, such as skin lesion disease diagnosis [52, 53]. Several self-supervised and semi-supervised learning-based medical image classification approaches are highlighted in [54, 55]. On the other hand, to ensure the privacy of data, Hossen *et al.* [56] propose a CNN-based federated learning framework for skin disease classification and ensuring Internet of Medical Things (IoMT) security. A personalized federated learning model can also be configured for a specific medical image analysis. For instance, in [57], the authors develop a personalized federated learning system to diagnose prostate cancer and classify skin lesion disease. Their investigation reveals that the performances of the client-specific different models vary significantly; because the datasets used to build the client-specific models may contain varying amounts of data that affect the bias of the trained federated learning framework.

2.2 Literature Review on Semantic Segmentation

Deep Learning-based Semantic Segmentation

Recent research efforts have extensively focused on deep convolutional neural networks (DCNNs) and their variants for semantic segmentation. These networks often use an encoder-decoder architecture, whereby the encoder decreases the spatial resolution of the input image while extracting an abstract representation [41]. Particularly convolutional neural networks have emerged as powerful tools for semantic segmentation, revolutionizing the field's ability to learn complex hierarchical representations directly from raw data. This thesis explores recent advancements and challenges in leveraging deep learning for semantic segmentation tasks. Due to DCNNs' robust feature learning capabilities, greater advancements have been achieved in image semantic segmentation tasks [58]. Among the existing works, fully convolutional networks [41, 58] laid an important foundation for deep learning-based image segmentation.

To address the challenges associated with reduced image resolution and the limitations of a neuron in a particular layer to capture sufficient context, U-Net-like architectures are considered one of the most suitable strategies for image segmentation. U-Net excels in medical image segmentation with the help of encoder-decoder concept [59]. Conversely, DeepLab [43] employs dilated convolutions and pyramid pooling as in PSPNet [46] to capture multi-scale contextual information. Mask R-CNN extends the Faster R-CNN framework to perform image segmentation, combining object detection and semantic segmentation [60]. The attention mechanisms commonly employed in natural language processing (NLP) [61] to model long-distance dependencies have been adapted for image segmentation tasks and have garnered considerable attention. For example, DANet [62] integrates an attention module into the ResNet backbone network, whereby parallel spatial and channel attention mechanisms capture long-range feature dependencies and improve segmentation accuracy. The local cross-channel interaction strategy was introduced for ECANet in [63]. The method's purpose is to attentively select the size of one-dimensional convolution kernels. GANs are another emerging approach that can be trained for image segmentation with fewer samples [64]. The PSPNet [46] has adopted parallel to construct an atrous spatial pyramid pooling (ASPP) model.

This module effectively captures contextual information at various scales, enhancing the quality of descriptors. Other architectures, such as DeepLab [43] and Furthermore, self-attention mechanisms, known for modeling feature dependencies, have garnered considerable attention.

2.2.1 Literature Review on GANs

Generative Adversarial Networks are a powerful tool in the field of generative modeling. Generative modeling is a type of machine learning where the goal is to learn the underlying distribution of a dataset and generate new samples that resemble the original data. Here, as a generative model GAN is capable of creating highly realistic data samples by leveraging the adversarial dynamics between the generator and discriminator. At the same time, GAN for semi-supervised learning is a powerful technique that leverages both labeled and unlabeled data to enhance the performance of machine learning models. In this framework, the GAN comprises two neural networks: a generator and a discriminator. The generator aims to create realistic data samples, while the discriminator tries to differentiate between real and generated samples. In semi-supervised learning, the discriminator is further extended to classify labeled data into different categories and identify whether an input sample is real or generated. This dual objective enables the discriminator to learn from both labeled and unlabeled data, using the unlabeled data to improve the feature representations and decision boundaries, thereby enhancing the model's generalization capability. This approach is particularly useful in scenarios where labeled data is scarce, allowing the model to make more accurate predictions and classifications by exploiting the abundance of unlabeled data.

Arguably Goodfellow *et al.* [65] pioneered GAN, a new class of generative models using adversarial processes. Luc *et al.* [66], then, advanced the GAN for the application of image semantic segmentation. This has been further extended to semi-supervised image segmentation scenarios by several researchers, like Hung *et al.* [67], and Li *et al.* [68]. It involves two distinct subnetworks—a generator and a discriminator. The generator learns from randomly distributed samples and produces synthetic data, $F(x)$. While, the discriminator assesses whether a given sample, either real, x or synthetic, $F(x)$=y, outputting the probability of its authenticity, as illustrated in Fig. 2.1.

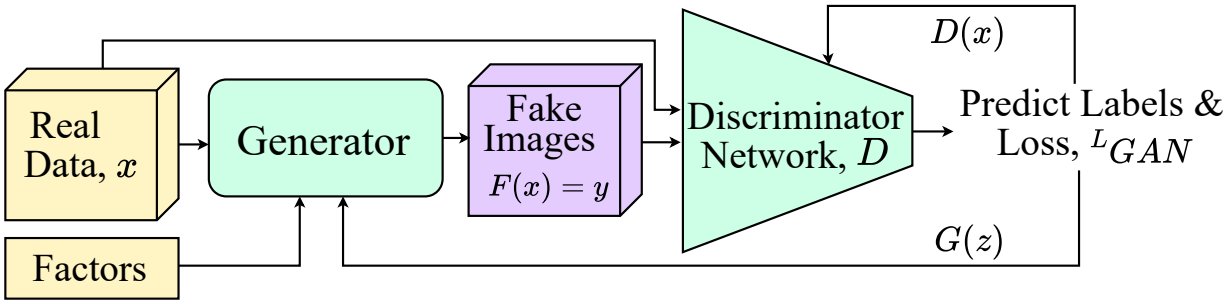


Figure 2.1: A Standard training procedure of GAN. The factors may differ from model to model w/t noise distribution, loss function, and optimization techniques used.

The loss function (2.1), $\mathcal{L}_{GAN}$ consists of the discriminator's loss $D(x)$, which measures the ability of the discriminator to distinguish between real and synthetic data, and generator's objective function $G(z)$, that evaluates the generator's strength in creating more realistic output.

$$\begin{aligned}\mathcal{L}_{GAN}(G, D) = \mathbb{E}_{x,y}[\log D(x)] \\ + \mathbb{E}_{x,z}[\log(1 - D(x, G(z)))],\end{aligned} \tag{2.1}$$

here $D(x)$ represents the discriminator's output for real data sample x, $G(z)$ is the generator's output when given input x and noise factor z, and $\mathbb{E}$ denotes the expected value. The discriminator works towards maximizing $\mathbb{E}_{x,y}[\log D(x)]$, the log probability of correctly identifying real data samples among generated samples, while the generator focuses on minimizing the log-probability $\mathbb{E}_{x,z}[\log(1 - D(x, G(z)))]$. Generator combines the losses from the discriminator and loss from itself to minimize the loss of the next step. Therefore, the discriminator can accurately identify the generated samples. Therefore, the adversarial training process involves the discriminator minimizing its loss while the generator simultaneously minimizes the GAN loss.

2.2.2 Literature Review on Semi-supervised Learning

Supervised learning approaches have played a dominant role in the machine learning paradigm for several decades. Yet, acquiring sizable labeled datasets is laborious and time-intensive. Consequently, the demand for models capable of learning from limited data is rapidly growing. In

response to these challenges, semi-supervised learning has appeared as an effective technique to address this issue [69]. Deep neural networks were employed to address semi-supervised semantic segmentation in [69]. Nevertheless, these approaches primarily concentrate on object segmentation, overlooking the broader semantic context and intricate interconnections within a scene. A selection mechanism was introduced in [70] by Nartey *et al.* with the goal of mitigating errors in reinforcement, a frequent issue encountered in traditional self-training models. They presented a generative semi-supervised learning model and designed to be robust against outliers and noisy samples. This model, leveraging a variational autoencoder (VAE), enhances its robustness by accounting for the uncertainty inherent in the input samples. Additionally, a de-noising layer is incorporated into the VAE architecture to further improve performance. Conversely, contrastive learning has been implemented recently in semi-supervised semantic segmentation methods with significant efficiency improvements. Zhou *et al.* [71] developed a cross-set region-level data augmentation technique to mitigate the feature inequality between labeled and unlabeled data. It is combined with cross-set pixel-wise contrastive learning to improve the model's feature representation capacity. Similarly, KE-GAN [72] captures semantic consistencies among different classes through a Knowledge Graph and incorporates a pyramid architecture when designing the discriminator to obtain multi-scale contextual information. Meanwhile, in [73], the s4GAN uses the segmentation network as the generator, with actual pixel-level annotations from labeled data and segmentation predictions from unlabeled data serving as inputs to the discriminator. The goal of this technique is to closely correlate the real annotated data with the prediction results of the unlabeled data.

Despite showing encouraging findings, the previous works have some drawbacks. Self-training and co-training approaches require repeated procedures to identify unlabeled samples, which leads to a longer execution duration. Furthermore, mislabeling any of these unlabeled data might negatively influence the generalization capacity of supervised learning models. This thesis aims to overcome these issues by devising a patch-wise discriminator and a self-gated activation-guided attention mechanism for the generator subnetwork.

Chapter 3

Self-supervised Image Classification

This chapter aims to grasp the basics of any non-supervised learning approaches and to explore how they differ from traditional supervised learning methods. In this direction, this chapter focuses on developing a self-supervised image classification model tailored for medical image classification, specifically for diagnosing Monkeypox (Mpox) from skin lesion images. Note that if a self-supervised model is fine-tuned on a small amount of labeled data after initial training on unlabeled data, it effectively becomes a semi-supervised model. Thus, building a self-supervised pipeline will lay the foundation for semi-supervised learning. Hence, the knowledge gained here will be instrumental in building the semi-supervised semantic segmentation model discussed in Chapter 5.

3.1 Overview

Mpox is a contagious viral illness that affects both humans and animals, with symptoms ranging from mild to severe, and its early diagnosis is critical for the effective management and prevention of this disease. The importance of timely identification is underscored by the potential for outbreaks, particularly in regions where healthcare resources may be limited.

One of the significant challenges in developing accurate diagnostic models is the requirement for large amounts of annotated data, typically provided by domain experts. However, the demand

for such expertise can be a bottleneck in rapidly evolving situations. The proposed method addresses this through a self-supervised learning framework, Simple Contrastive Learning for Representations (SimCLR), which is an efficient model to extract general discriminative patterns from unlabeled data and can be applied to tasks like skin lesion classification, including Mpox. SimCLR helps the model to learn robust representations by contrasting different augmented views of the same image, enhancing its ability to differentiate between Mpox lesions and other skin conditions.

Federated learning (FL), on the other hand, is integrated into the proposed model to enable a privacy-preserved collaborative training environment. FL allows the Mpox classifier to be built on vast and diverse datasets collected from several healthcare institutions across different geographical regions, without the need to centralize sensitive patient data. This decentralized approach not only preserves patient privacy but also facilitates the inclusion of data from a wider array of sources, improving the model's generalizability and robustness. By training on diverse datasets remotely, the model becomes more adaptable to varying presentations of Mpox, making it a valuable tool in global health initiatives.

3.2 Monkeypox Diagnosis

Monkeypox is a viral disease caused by the monkeypox virus. As it has the potential for human-to-human transmission and affects both humans and animals, it poses a significant public health concern. As per the report of Disease Control and Prevention (CDC), there is no suitable treatment available for the Monkeypox virus [74]. However, the CDC has authorized two oral drugs, Brincidofovir and Tecovirimat, which were primarily utilized in treating the smallpox virus, for the treatment of Mpox. Therefore, early detection and accurate diagnosis are crucial for effective treatment planning, disease control, and prevention [75].

Nevertheless, the clinical features of Mpox can be similar to other diseases, such as chickenpox and smallpox, making an accurate diagnosis of Mpox challenging, especially in regions with limited access to skilled healthcare professionals and diagnostic facilities [76], [77]. It urges the

Table 3.1: Existing survey of monkeypox detection and classification.

Models	Objectives	Dataset	Limitations
CNN+Federated [56]	Multi-class classification of human skin diseases.	DermNet Database	This model does not measure the severity of the diseases. Small amount of data.
DenseNet-201 [81]	Develop a new dataset of monkeypox with four classes. Multi-class Monkeypox identification using the DL model.	MSID	An imbalanced dataset.
VGG16[82]	Binary classification to detect Monkeypox.	Monkeypox2022	Followed a similar data organization pattern like MSID. But the dataset has very limited samples and is smaller than the MSID dataset.
MobileNetV2 [83]	Developed a small dataset. After that, evaluate the model with the MSLD dataset.	Data_monkeypox (Kaggle)	Only 117 samples were used to classify the monkeypox. Where 45 samples represent monkeypox.
ResNet50 [1]	An AI-assisted diagnosis system implemented to detect monkeypox.	MSLD	a small dataset for binary classification.
MobileNetV2, EfficientNetb0 [84]	Developed an android application to visualize monkeypox detection results.	MSLD	A Small dataset for binary classification.
Ensemble Model [48]	Detect monkeypox for binary classification.	MSLD	A small dataset for binary classification. Only apply Gaussian noise augmentation.

research community to develop novel alternative methodologies for the aforesaid diagnosis. However, such developments must meet regulatory compliance, like patient privacy, and data security. Federated learning provides a distributed and collaborative machine learning approach without sharing of patient's raw data [78]. In the context of Mpox diagnosis, the federated learning framework can leverage data from multiple healthcare facilities or regions, allowing the development of robust and generalized classifiers [79, 80]. A general summary of recent existing models, datasets, and limitations related to monkeypox classification is listed in Table 3.1.

Recently, successful deep learning models for image classification tasks are largely contingent on the availability of high-quality annotated samples [85], which can be scarce and challenging to acquire in the clinical field, particularly for rare diseases like Mpox. But the self-supervised learning has the ability to extract the intrinsic structure of unlabeled data, mitigating the dependency on large-scale labeled datasets. Thus, it is a natural progression to exploit the best of the worlds of FL

and self-supervised learning, enabling rapid and accurate diagnosis, and assisting in public health interventions.

3.3 Methodology

Fig. 3.1 illustrates the overview of the proposed methodology.

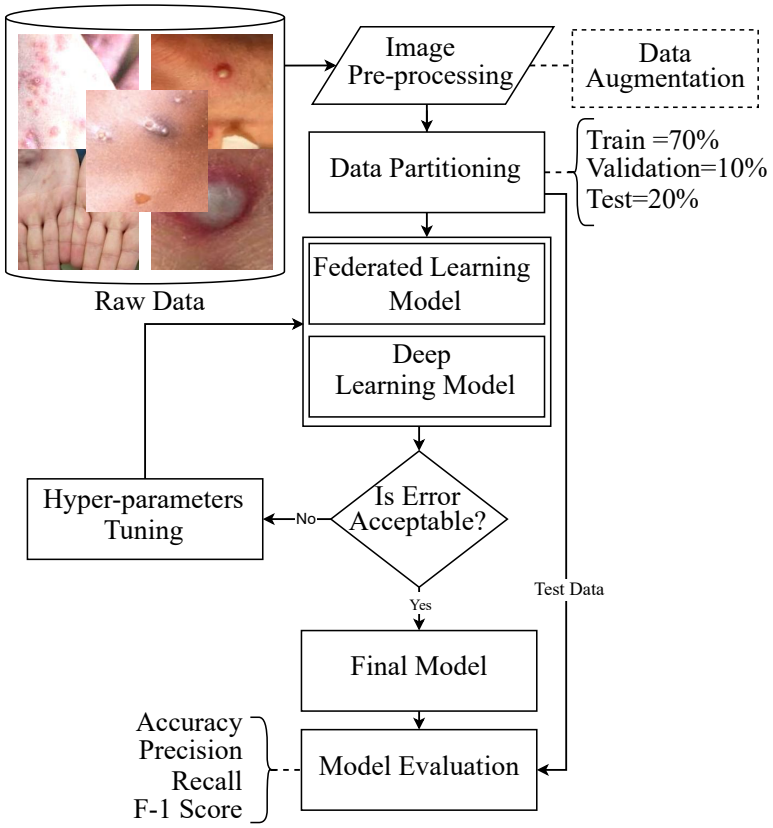


Figure 3.1: The workflow of the proposed methodology for monkeypox classification using federated learning framework with semi-supervised and self-supervised approaches.

3.3.1 Benchmark Dataset

Samples from the Monkeypox Skin Lesion (MSL) dataset [1] are used for model building and validation. The dataset comprises a total of 228 samples of the size of 224×224 with annotations: Monkeypox and other. Specifically, the Monkeypox class includes 102 samples, while the other

class contains 126 samples. From these samples, a mutually exclusive set of training, validation, and test sets are created with a 70:10:20 ratio. Fig. 3.2 shows eight random samples from the dataset representing the two classes.

Figure 3.2: A collection of random samples from the MSL [1] dataset.

3.3.2 Image Pre-processing

In the realm of medical imaging-based prognoses and diagnoses, data pre-processing holds significant importance. Here, the application of data augmentation techniques to raw images proves crucial. By employing fifteen distinct transformations, as depicted in Fig. 3.3, the raw images are synthetically transformed to generate a diverse set of training samples. This approach not only enhances dataset diversity but also boosts a model's generalization capabilities.

- **Rotation:** Rotating the image by a certain angle. Here, the augmented image is rotated by $90°$ and between $-45°$ to $-45°$.
- **Contrast:** Increasing contrast (Value = 2.5) enhances the separation between different image elements, making the image appear more vibrant and visually appealing.

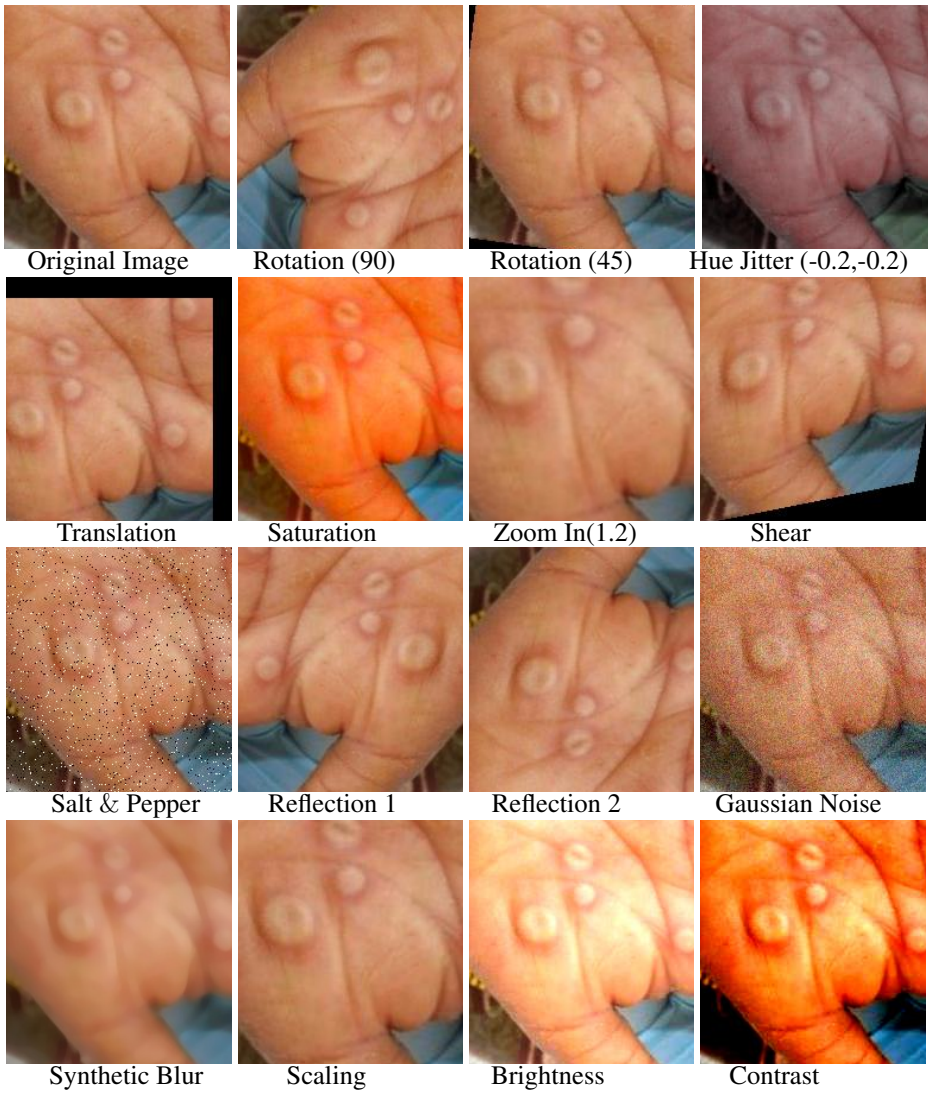


Figure 3.3: A group of samples generated through augmentation. The top-left image is an original sample provided to understand the variation created by the data augmentations.

- **Brightness:** It refers to the overall intensity of light in an image. It determines how light or dark an image appears. Here, the brightness factor is 0.5.

- **Gaussian Blur:** This is a popular image filtering technique that applies a blur effect to an image. It uses a Gaussian distribution to determine the amount of blurring at each pixel. Gaussian blur smooths out high-frequency details, reducing image noise and creating a softer appearance. The standard deviation needed to produce Gaussian noise is 20.

- **Hue Jitter:** To introduce random variations in the hue (color) of an image. It involves applying small, random shifts to the hue values of pixels. A natural variation in artistic effects by using min_jitter = -0.2 and max_jitter = 0.2.

- **Reflection:** It refers to the visual effect of an object being mirrored or duplicated on a reflective surface. In image processing, reflection can be applied to create a mirror-like (vertical and horizontal) appearance or simulate the reflection of an object on a shiny surface.

- **Saturation Jitter:** This augmentation helps to enhance or reduce the intensity of colors, making the model more robust to variations in color saturation. In this study, the saturation range is 2 to 2.5.

- **Translation:** It involves shifting the image's content horizontally and vertically by a certain number of pixels, the range was -50 to 50 (in pixels). By introducing random translations, the model learns to recognize objects at different positions in the image, improving its ability to generalize to unseen data.

- **Salt and Pepper Noise:** In this augmentation, random pixels in the image are set to either maximum intensity (salt) or minimum intensity (pepper). In both cases, the salt and pepper probability was 0.02. It helps the model become more resilient to noisy input and aids in training it to handle real-world scenarios with varying levels of noise.

- **Synthetic Blur:** Synthetic Blur is an augmentation technique used to replicate blurriness in images. For creating a blur effect, apply median blur with a kernel size of 11.

- **Scaling:** Scaling is a transformation applied to images by resizing them to different dimensions, such as 300×300. Both up-scaling and down-scaling can be used as data augmentation techniques.

- **Shear:** Shear augmentation involves altering the shape of an image by slanting or skewing it along either the horizontal or vertical axis using a shear factor (0.2). This creates a transformed version of the original image that appears as if viewed from a different angle, enhancing the robustness of machine learning models to various perspectives.

3.3.3 The proposed DL Model

CNN for Mpox Classification

CNNs for image classification are composed of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers apply a set of learnable filters to extract local patterns such as edges, textures, and shapes, by sliding the filters across the image and computing the dot product between the filter and input patches. The pooling layers reduce the spatial dimensions of the features, decreasing computational load and helping to make the representations invariant to small transformations and distortions. Common pooling operations include max pooling and average pooling, which summarize regions of the feature map. The extracted features are then fed into fully connected layers to perform classification based on the high-level representations. This hierarchical structure of CNNs enables them to automatically learn complex patterns and relationships in the data, making them highly effective for image classification tasks.

Table 3.2: Layer-wise details of the proposed DL Model

Layer	**Size**	**Filter Size**	**Stride**	**Activation**
Input	(224, 224, 3)	-	-	-
Convolutional	(224, 224, 256)	3×3 256 filters	1	ReLU
Max Pooling	112, 112, 256)	2×2	2	-
Convolutional	(112, 112, 128)	3×3 128 filters	1	ReLU
Max Pooling	(56, 56, 128)	2×2	2	-
Convolutional	(56, 56, 64)	3×3 64 filters	1	ReLU
Max Pooling	(28, 28, 64)	2×2	2	-
Convolutional	(28, 28, 32)	3×3 32 filters	1	ReLU
Max Pooling	(14, 14, 32)	2×2	2	-
Dropout	(14, 14, 32)	-	-	-
Flatten Layer	(6272)	-	-	-
FC Dense_1	(512)	-	-	ReLU
FC Dense_2	(256)	-	-	ReLU
Dense_3	(2)	-	-	Sigmoid

Learning rate (l_r) = 0.001, Batch size = 32, Optimizer = Adam,
Objective function = BCE, Number of trainable parameters = 1,838,706.

Table 3.2 provides the architectural details, including the convolution (Conv) layers with their respective output feature maps, pooling layers, activation function, and kernel size of the respective

layers. The convolution layers learn robust invariant features that are specific to Mpox from skin lesion images using 2-D convolution operations C, as given in (3.1).

$$C(m,n) = b + \sum_{k=0}^{K-1}\sum_{l=0}^{K-1} f(k,l) * x(m+k, n+l), \tag{3.1}$$

where f is the filter, b is the bias, and x is the input. Hence, $*$, K, $\{m,n\}$, and $\{k,l\}$ denote convolution operation, filter size, input origin, and element index of the filter respectively. The network rectifies the outputs of each convolution operation through a rectified linear unit (ReLU) defined as $max(c_{\{m,n\}}, 0)$, where $c_{\{m,n\}}$ is a value in the output feature map of a respective convolution. To extract key feature values from the convolution feature maps and reduce the dimensionality, the model employs the max pooling operation as defined in equation (3.2).

$$\text{maxpool}(x) = \max\left(x_{\{i:j\}}\right), \tag{3.2}$$

where x represents the input feature map, while i and j denote the start and end indices of the pooling region, respectively. After the third max pooling operation, a flattening layer is introduced to convert the 2-D feature maps into a 1-D vector that facilitates connectivity toward the densely connected classifier at the top. Finally, two fully connected layers with ReLU activation functions, and an FC with Sigmoid activation as expressed in (3.3) to perform the required classification accurately.

$$\sigma(x) = \frac{1}{1+e^{-x}}, \tag{3.3}$$

where x and $\sigma(x)$ represent the input vector, and the probability of input to be positive class is Mpox. Note that before the flattening operation, a dropout of 0.25 is applied to reduce the interdependence of neurons by randomly deactivating a given percentage of neurons, which helps the model to fight against overfitting issues. Also, these connections have associated weights (W) that the network adjusts during training to learn patterns in the input data. This CNN architecture is

considered for both supervised and self-supervised approaches during the training phase elaborated in Section 3.3.4.

The Federated Learning Framework

This learning approach is particularly valuable in scenarios, where data cannot be centrally aggregated due to privacy concerns. This property is crucial for dealing with Mpox, as the disease can emerge in different regions with distinct visual manifestations. A Federated Learning Framework for medical image classification enables multiple healthcare institutions to collaboratively train a machine learning model without sharing their sensitive patient data. Each participating institution, acting as a local node, trains the model on its private data and periodically sends model updates to a central server. This server aggregates the updates using techniques such as Federated Averaging, improving the model while preserving individual data privacy. The framework is designed to address challenges like data heterogeneity across different hospitals, ensuring the model can be generalized across diverse patient populations and imaging devices. Advanced privacy-preserving methods, such as differential privacy and secure multiparty computation, are employed to further safeguard patient information. By using efficient communication strategies, including compression techniques and asynchronous updates, the framework minimizes the data transfer load, facilitating real-world deployment across varying network conditions and computational resources.

Fig. 3.4 illustrates the FL process begins with the initialization of a global model (G_m) at a central server. The participating clients independently train their local models on their respective data samples without sharing raw data with other clients. This training can involve any machine learning techniques, including deep learning, here, it is a CNN (Section 3.3.3). The aggregation function to integrate the learned weights at the clients and to update the G_m is defined in (3.4).

$$W_t = \sum_{i=1}^{n} \frac{l_i^t}{l_t} \cdot W_i^t, \tag{3.4}$$

where t and i stand for the current training iteration and the local client's model, respectively. Hence, W_t is the updated weight parameter of the global model G_m, l_t is the global model's

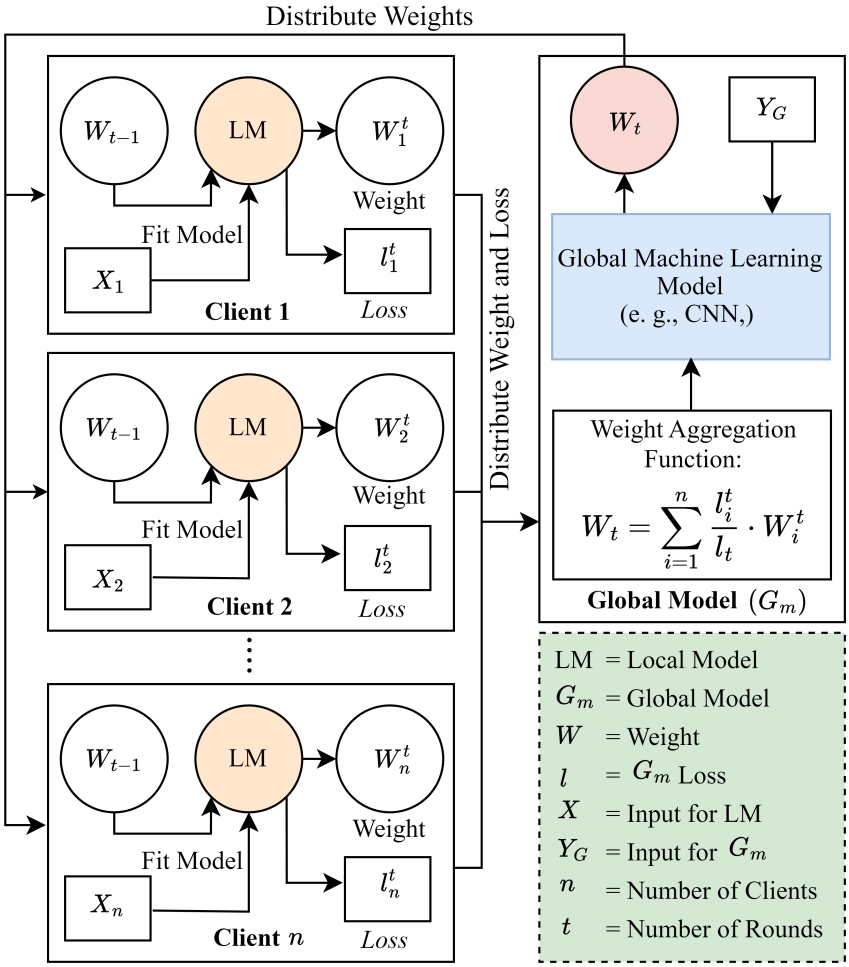


Figure 3.4: A general illustration of FL architecture. The global model aggregates the learned weights from the clients to update the model's weight. This new weight is shared with the clients to update the local models for refinement.

current loss, l_i^t is the current loss of the local model, W_i^t is the local model's current weight, and n is the total number of clients participating in the FL framework. In this thesis, due to the lack of computational resources n is set to 5, and the total number of training iterations, t is set to 15.

3.3.4 Training Strategy

Supervised Training

The first CNN is built following a supervision learning approach assisted by the FL framework. Here the proposed CNN is trained using an Adam optimizer with a learning rate (l_r) of 0.001 by

feeding a mini-batch (b) of 32 samples from the training set (MSLD) to minimize the mini-batch binary cross-entropy loss defined by (3.5).

$$E = \frac{-1}{n} \sum_{n=1}^{b} [p_n \log \hat{p}_n + (1 - p_n) \log(1 - \hat{p}_n)], \tag{3.5}$$

where it takes two arguments: $\hat{p}$–the output from the top layer of the CNN (cf. Table 3.2), and p_n– the target ($p_n \in [0, 1]$). At the end of the 25th epoch, the model is converged as one can observe the training progress shown in Fig. 3.6 (a) on page no. 43.

Self-Supervised Training

The second CNN is trained using a SimCLR-based SSL technique under the same FL framework elaborated in Section 3.3.3. SimCLR is a self-supervised learning approach designed to learn useful visual representations without requiring labeled data. Developed by Google Research, SimCLR leverages contrastive learning to train deep neural networks by maximizing the similarity between different augmented views of the same image while minimizing the similarity between views of different images. The approach involves two key components: a base encoder network and a projection head. The encoder network, typically a convolutional neural network, extracts features from the input images. Here, the SimCLR works as a pretext task followed by the Mpox classification performed by the proposed CNN. The SimCLR begins input feature embedding through an encoder network (the backbone network is a ResNet) followed by a multi-layer perceptron projection head that maps the embedding into a lower-dimensional space. Fig. 3.5 depicts the basic structure of the SimCLR and equation (3.6) express the contrastive loss function [86] that indicates positive pair of samples (i, j) used in SimCLR.

$$l_{(i,j)} = -log \frac{exp(S_{i,j})}{\sum_{k=1}^{2N} l_{[k!=i]} exp(S_{i,k})}, \tag{3.6}$$

where N is the batch size, i, and j is the augmented images of the same image, and k is the negative sample image. Therefore, $S_{i,j}$ is for the positive sample and $S_{i,k}$ for the negative sample. To find

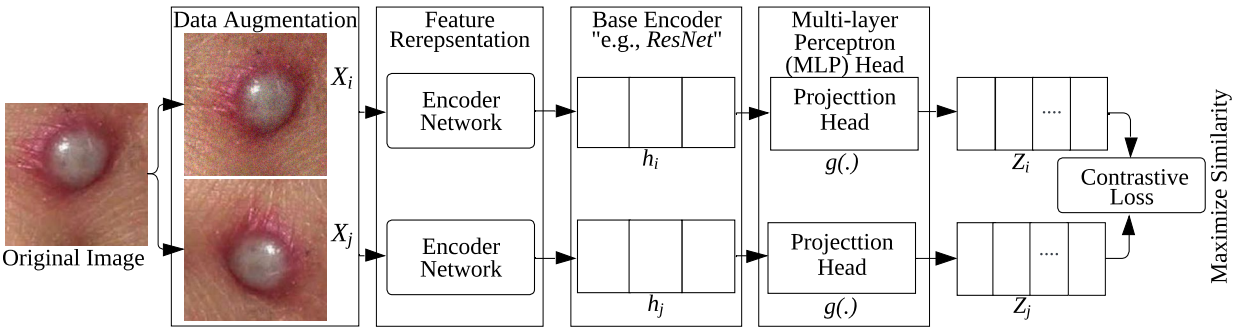
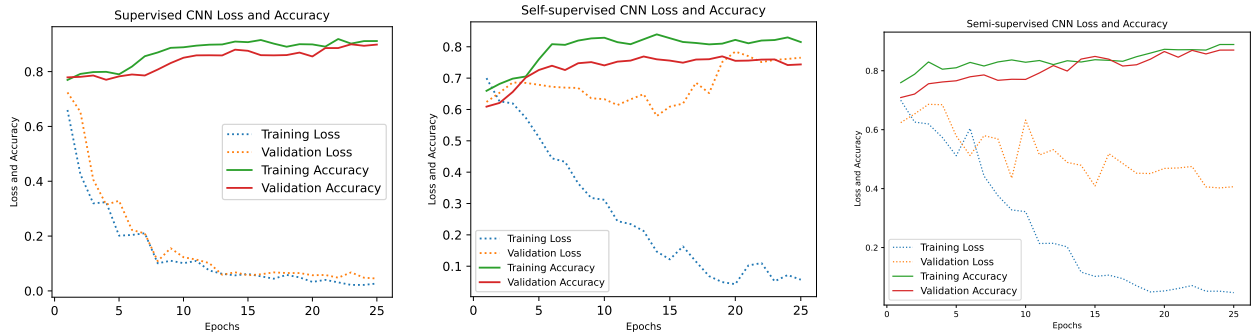


Figure 3.5: An illustration of the SimCLR architecture. ResNet works as an encoder network and a contrastive loss $l_{(i,j)}$ decides the performance of the SimCLR.

the similarity matrix and for getting positive and negative pairs the symbol $l_{[k!=i]}$ represents the condition that K is not equal to i and this is working as an indicator function.

To evaluate the supervised and self-supervised approaches, the training versus validation accuracy and loss during the training phase are monitored as shown in Fig. 3.6.



(a) (b) (c)

Figure 3.6: Training progress of two different learning approaches: (a) supervised, (b) self-supervision, (c) semi-supervision.

Initially, during the early stage of the training, the training loss exhibits a steep decline, indicative of the model quickly adapting to the training data. Simultaneously, the validation loss demonstrates a parallel reduction, underscoring the model's capacity to generalize beyond the training set. The same observation is also applicable for training and validation accuracy.

Semi-Supervised Training

In this study, a semi-supervision framework was implemented to enhance the binary classification of monkeypox skin lesion images. The training process began with the proposed CNN architecture designed for image classification. Initially, the model was trained using a 30% labeled and 70% unlabeled dataset of images, where each image was manually annotated as either 'monkeypox' or 'Other'. The model generated the predicted (pseudo) label for the unlabeled dataset.

Subsequently, the trained model can generate predictions on a larger set of unlabeled data, which contains images without any pre-assigned labels. A label prediction technique was applied by assigning labels to the unlabeled images based on the model's predictions, selecting only those predictions that exceeded a confidence threshold of 50% to ensure label accuracy and minimize noise. The pseudo-labeled data were then combined with the original labeled dataset, creating an expanded training set that provided a richer source of information for the model. The CNN model was retrained using this combined dataset, allowing it to refine its feature extraction and classification capabilities. Throughout the training process, the model's performance was monitored using a validation set, and key metrics such as accuracy, precision, recall, and F1-score were used to evaluate its effectiveness. This semi-supervised training process not only leveraged the limited available labeled data but also capitalized on the large volumes of unlabeled data, ultimately enhancing the model's ability to generalize and accurately classify monkeypox lesions.

3.4 Experimental Results

3.4.1 Experimental Setup

To speed up the FL-assisted training procedure, an adequate storage capacity is essential on both the central server and edge devices to accommodate the datasets for model training. In this case, the proposed model developed developed with Python Ver. 3 and DL libraries, such as Keras and TensorFlow, and experimented on a computing platform with a Tesla T4 GPU with 12GB memory.

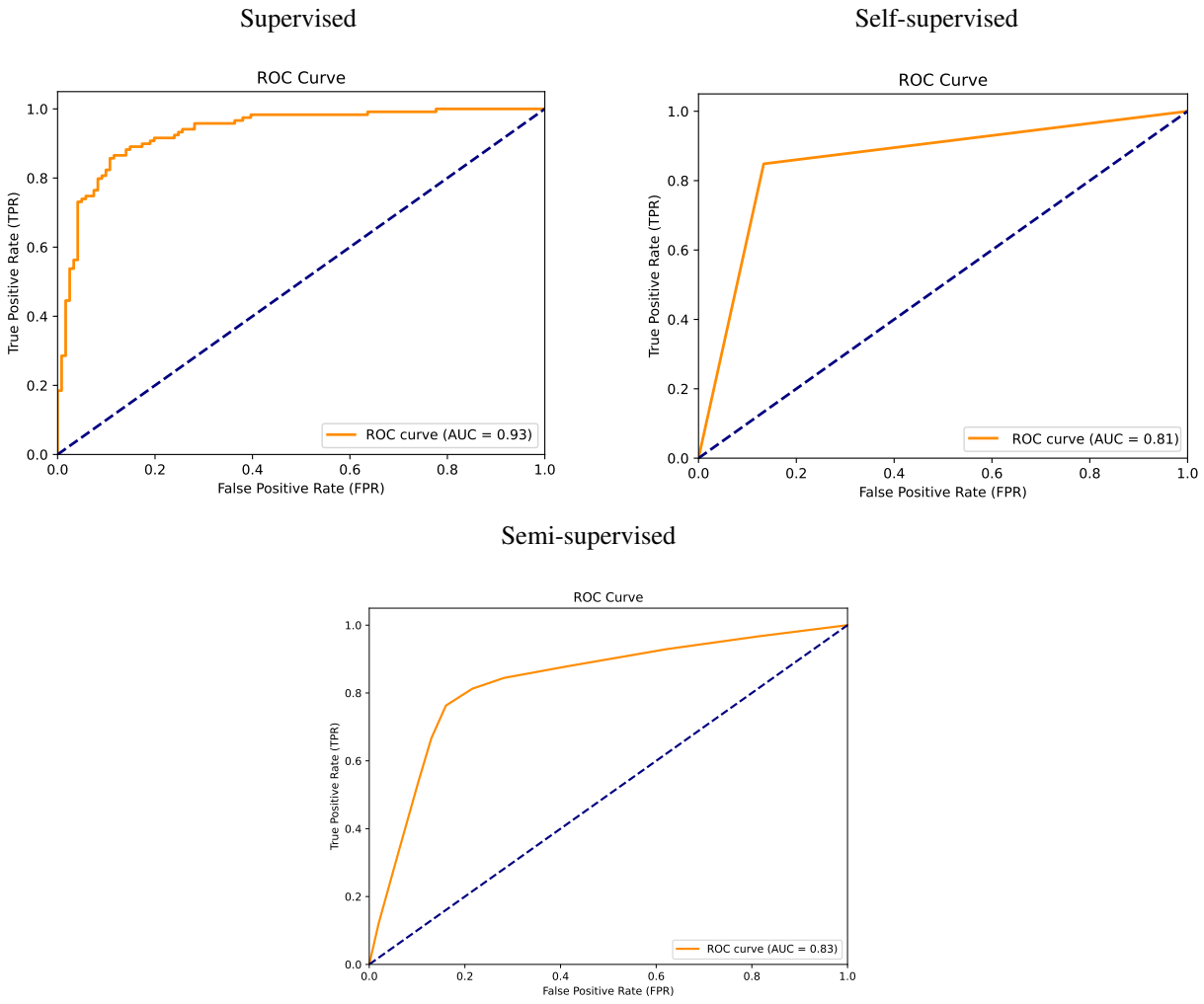


Figure 3.7: AUC-ROC of the proposed supervised, self-supervised and semi-supervised CNN on FL framework.

3.4.2 Quantitative Analysis

$$\text{Accuracy} = \left(\frac{\text{Number of correctly classified images}}{\text{Total number of images}} \right) \times 100 \tag{3.7}$$

Table 3.3 on page no. 46 compares the performance of the proposed approaches with important few existing works where the comparison metric is accuracy. Accuracy for images refers to the percentage of correctly classified images out of the total number of images in a dataset defined by equation (3.7). On the other hand, Fig. 3.7 shows the performance of the proposed model in terms of AUC-ROC curve. This curve combines two components: ROC and AUC curve. The

Table 3.3: Comparison of the proposed model with existing solutions for monkeypox binary classification task on MSL dataset [1]. Note: FL - Federated learning, DA - Data augmentation, NA - Not available in the literature.

Approach	# of DA	Input Dimension	Dropout	Optimizer	GFLOPs	Accuracy
Ensemble Model [48]	10	(224, 224, 3)	NA	NA	NA	93.4%
ResNet50 [1]	13	(224, 224, 3)	0.3, 0.2	Adam	NA	88.0%
EfficientB0 + DenseNet [50]	NA	(224, 224, 3)	NA	NA	NA	94.6%
MiniGoogLeNet [51]	8	(224, 224, 3)	NA	NA	NA	97.1%
Proposed (FL + supervised CNN)	15	(224,224, 3)	0.25	Adam	2.99	90.0%
Proposed (FL + self-supervised CNN)	15	(224,224, 3)	0.25	Adam	1.42	81.7%
Proposed (FL + semi-supervised CNN)	15	(224,224, 3)	0.25	Adam	2.48	82.9%

ROC curve, which plots the true positive rate (sensitivity) against the false positive rate at various decision thresholds, and the AUC, which represents the area under the ROC curve. The AUC value ranges from 0 to 1, with 1 indicating a perfect classifier, 0.5 indicating random guessing, and values below 0.5 implying worse-than-random performance. A higher AUC score means the model has a better ability to distinguish between positive and negative classes across different thresholds. For the monkeypox classification problem, the supervised CNN achieves a 93.00% AUC and 90.00% accuracy, the self-supervised CNN records an 81.00% AUC and 81.67% accuracy and finally, semi-supervised approach achieves an 82.90% accuracy with 83.00% AUC. Although these results are lower than the best of the existing solutions, it must be noted that beyond the imperfection in the accuracy, the proposed solutions in this work address the key issue of patient data privacy through federated learning and handle the annotated data scarcity via self-supervision and semi-supervised approach. Traditional supervised learning requires huge label data to get promising results, this experiment investigates the impact of self-supervision and semi-supervision for image classification tasks.

3.5 Conclusion

This work introduces a pioneering approach for Mpox classification, utilizing federated learning in conjunction with a CNN model. The CNN model incorporates both supervised, self-supervised, and semi-supervised learning techniques. This method employs federated learning to ensure pri-

vacy and uses SimCLR as a pretext task to extract feature maps from unlabeled samples. Some of the existing works report to have a better performance than this work. However, according to our survey, this is the first time FL has been exploited for Mpox diagnosis. With the consideration of data privacy as a major concern, the FL approach provides the novelty of this thesis, and that makes the results obtained in this experiment acceptable. Furthermore, semi-supervised learning outperforms the self-supervised learning approach. Therefore, this thesis contributes considerably to the development of advanced diagnostic systems for Mpox, enabling early detection and efficient disease management. Future research endeavors aim to expand this approach to larger-scale federated learning frameworks and explore various self-supervised pretext tasks and semi-supervised frameworks to further enhance the performance of the proposed model.

Chapter 4

Developing a Binary Segmentation Model: the Foundation of Semantic Segmentation

This chapter aims to establish the fundamentals of developing and evaluating a deep learning-based image segmentation model. It investigates and implements a convolutional neural network-based binary segmentation model to accurately detect and localize pavement cracks in road scene imagery. Thus, it lays a strong foundation for advancing toward a semi-supervised semantic segmentation model in the next stage (cf. Chapter 5).

4.1 Pavement Crack Segmentation

Monitoring pavement conditions is pivotal in managing road assets and ensuring the structural integrity and reliability of highways amidst Canada's diverse environmental conditions. Early detection of cracks serves as a primary indicator of pavement deterioration, where timely repairs curtail maintenance expenses, prolong infrastructure lifespans, diminish fuel consumption, and enhance safety and ride comfort. Various factors, including severe weather fluctuations due to climate change, natural aging of roads, and escalating heavy traffic loads, contribute to pavement crack formation. In 2021, the Canadian Automobile Association (CAA) highlighted that inadequate road quality costs drivers an additional $126 per vehicle annually, amounting to a staggering

$3 billion nationwide. Ontario alone bears a significant burden, accounting for $750 million of this sum [87]. The research underscores the dual financial impact on citizens caused by deteriorating roads: increased vehicle operation expenses and escalated government expenditure on infrastructure repairs. Given Ontario's extensive network of approximately 42,000 kilometers of two-lane highways and its pioneering initiative in implementing 2+1 highways. Thus, regular pavement assessments are imperative to deploy maintenance crews promptly and prevent irreparable damage. Yet, manual year-round monitoring and pavement assessments entail substantial labor, resources, and time. Addressing these challenges requires implementing a robust automated assessment system. Investing in artificial intelligence and computer vision-driven solutions for pavement analysis today could potentially prevent or delay hefty expenditures on future rehabilitation or reconstruction projects. Although the Ministry of Transportation of Ontario (MTO) currently employs Automatic Road Analyzer (ARAN) vehicles to scan the entire highway network, there remains an opportunity for further research into complementary technologies to develop robust models for detecting and predicting pavement deterioration based on pavement section images and meteorological data for specific locations or zones. Meanwhile, image segmentation has been a cornerstone for various applications, including pavement management systems (PMS). As an essential process of partitioning a digital image into multiple segments, it facilitates the simplification or change of an image's representation into something more readable and more accessible to high-level analysis, viz., object detection, recognition, or scene understanding. Meanwhile, the efficiency of the segmentation directly impacts the performance of the higher-level tasks, making it a critical area of research in computer vision. For example, pavement crack segmentation is one of the most demanding fields for regularly maintaining pavement safety and highway infrastructure. However, a high-precision segmentation model is required to identify and assess the extent of pavement damage.

Many recent studies have claimed that integrating deep neural networks has significantly advanced the capabilities of image segmentation models [88, 89]. For instance, the U-Net [38] architecture has gained prominence for its effectiveness in image segmentation tasks. It was originally developed for biomedical image segmentation but later adapted for other image segmentation tasks,

including road crack detection. Meanwhile, pavement surfaces present complex patterns and textures where cracks vary in size, shape, and visibility. Like U-Net's ability to capture fine-grained details through its multi-scale contextual information processing, the DNNs are invaluable in this context [90, 91].

In this direction, this chapter discusses the implementation of an attention mechanism (a strategy that allows DL models to focus on the most critical features of the input data, improving their ability to capture complex relationships and patterns) and a self-gated activation-driven pavement crack segmentation model. The main contributions of this work are listed below:

- It introduces an optimized architecture with an attention block and self-gated activation to improve pavement crack segmentation accuracy compared to existing counterparts.
- It conducts extensive experiments on two benchmark datasets and systematically compares the proposed model with recent pavement detection methods to validate the proposed model's performances.

4.2 Methodology

This section discusses the dataset, the proposed model, and the training process. To tackle the task of pavement crack detection, it elaborates on the standard U-Net model to enhance its efficiency.

4.2.1 The Proposed Architecture

Table 4.1 describes the layer-wise connectivity pattern of the proposed pavement crack segmentation model. The baseline and proposed models' input layer is configured to accept a visible spectrum (RGB) image as input of 256×256. Hence, their encoding and decoding sub-networks remain the same, where they use a standard convolution with a kernel of size 3×3, stride rate of 1, and padding to be `'same'` followed by batch normalization (BN) and ReLU operations. The encoding sub-network uses max-pooling layers with a stride of 2 to down-sample the spatial

Table 4.1: The layer-wise architectural description of the proposed segmentation model.

Proposed Model with Attention Block and Self-gated Activation				
	Layer ID	Layer type $A(k, s)$	Output Shape $[b, H, W, D]$	Input
Encoding phase	Input	Input Layer	$[b, 256, 256, 3]$	mini-batch
	L1	Conv (3, 1)$\rightarrow$ ReLU	$[b, 256, 256, 32]$	Input
	L2	Conv (3, 1)$\rightarrow$ ReLU	$[b, 256, 256, 32]$	L1
	L3	MaxPooling (2, 2)	$[b, 128, 128, 32]$	L2
	L4	Conv (3, 1)$\rightarrow$ ReLU	$[b, 128, 128, 64]$	L3
	L5	Conv (3, 1)$\rightarrow$ ReLU	$[b, 128, 128, 64]$	L4
	L6	MaxPooling (2, 2)	$[b, 64, 64, 64]$	L5
	L7	Conv (3, 1) $\rightarrow$ ReLU	$[b, 64, 64, 128]$	L6
	L8	Conv (3, 1)$\rightarrow$ ReLU	$[b, 64, 64, 128]$	L7
	L9	MaxPooling (2, 2)	$[b, 32, 32, 128]$	L8
	L10	Conv (3, 1)$\rightarrow$ ReLU	$[b, 32, 32, 256]$	L9
	L11	Conv (3, 1)$\rightarrow$ ReLU	$[b, 32, 32, 256]$	L10
	L12	MaxPooling (2, 2)	$[b, 16, 16, 256]$	L11
	L13	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 512]$	L12
	L14	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 512]$	L13
Attention Block	L15	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 256]$	L14
	L16	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 256]$	L15
	L17	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 256]$	L16
	L18	Add	$[b, 16, 16, 256]$	L16, L17
	L19	Conv (3, 1)$\rightarrow$ ReLU	$[b, 16, 16, 256]$	L18
	L20	Conv (3, 1)$\rightarrow f(\cdot)$	$[b, 16, 16, 1]$	L19
	L21	Up-sample (2, 2)	$[b, 32, 32, 1]$	L20
	L22	Multiply	$[b, 32, 32, 256]$	L21, L11
	L23	Conv (3, 1)$\rightarrow$ ReLU	$[b, 32, 32, 256]$	L22
Decoding phase	L24	Up-sample (2, 2)	$[b, 32, 32, 512]$	L23
	L25	Cat	$[b, 32, 32, 768]$	L24, L23
	L26	Conv (3, 1)$\rightarrow$ ReLU	$[b, 32, 32, 256]$	L25
	L27	Conv (3, 1)$\rightarrow$ ReLU	$[b, 32, 32, 256]$	L26
	L28	Up-sample (2, 2)	$[b, 64, 64, 256]$	L27
	L29	Cat	$[b, 64, 64, 384]$	L28, L8
	L30	Conv (3, 1)$\rightarrow$ ReLU	$[b, 64, 64, 128]$	L29
	L31	Conv (3, 1)$\rightarrow$ ReLU	$[b, 64, 64, 128]$	L30
	L32	Up-sample (2, 2)	$[b, 128, 128, 128]$	L31
	L33	Cat	$[b, 128, 128, 192]$	L32, L5
	L34	Conv (3, 1)$\rightarrow$ ReLU	$[b, 128, 128, 64]$	L33
	L35	Conv (3, 1)$\rightarrow$ ReLU	$[b, 128, 128, 64]$	L34
	L36	Up-sample (2, 2)	$[b, 256, 256, 64]$	L35
	L37	Cat	$[b, 256, 256, 96]$	L36, L2
Top	L38	Conv (3, 1)$\rightarrow$ SiLU	$[b, 256, 256, 32]$	L37
	L39	Conv (3, 1)$\rightarrow$ SiLU	$[b, 256, 256, 32]$	L38
	L40	Conv (1, 1)$\rightarrow f(\cdot)$	$[b, 256, 256, NC]$	L39
Total number of trainable parameters				8,379,140
$A(k, s)$: A- operation type, k - kernel size, and s - stride rate; Output shape as $[b, H, W, D]$: b - mini-batch size, H - height, W - width, and D - number of channels; $f(\cdot)$ - classifier (Sigmoid), NC - number of output channels				

dimensions while returning essential features. The decoding sub-network employs an interpolation (`nearest neighbor`)-based upsampling operation to increase the spatial dimensions of the input feature maps. However, the proposed network's bottleneck and top layers are differently structured compared to the baseline to better capture the intricate features of the cracks from the pavement visuals. The proposed architecture interspaces an attention block (`L15 - L23`) between encoding and decoding sub-networks. This block is a sophisticated gating mechanism to emphasize the salient features. The top layer in the proposed model exploits the SiLU activation instead of ReLU; such systematic changes boost the generalized performance of the model. Finally, the output layers of both architectures employ a convolution operation with a kernel of size 1×1 and generate a pavement crack probability map predicted by a `sigmoid` activation.

The Attention Sub-Network

The attention mechanism in DNNs, coupled with its effectiveness in natural language processing, has achieved significant progress in computer vision, including semantic segmentation [92, 93]. The studies show that the attention mechanism can locate pixels that hold crucial contextual information for better visual recognition capability. According to the studies, pixels containing crucial contextual data can be identified by the attention mechanism, which improves visual recognition.

The attention layer in U-Net involves the calculation of the attention features map and is successful in computer vision, such as image classification and segmentation. It performs feature concatenation and element-wise addition ($\oplus$) to get a rich feature map. The sigmoid function σ generates a gated weight and performs element-wise multiplication ($\odot$) with the input feature X to generate the refined attention feature map X' as defined by (5.4).

$$X^{'} = (X \odot \sigma(f_g \oplus f_x)), \tag{4.1}$$

where the feature map is calculated using f_g ($[b_g H_g W_g D_g]$) and f_x ($[b_x H_x W_x D_x]$). These are outputs of upsampling and encoder operation, and they follow element-wise addition. After that, the σ operation generates the final feature map by using dot-product. The attention block contributes to

the model's ability to adaptively weigh the importance of different spatial information, leading to more precise and context-aware segmentation results. Thus, this chapter incorporates an attention sub-network as shown in Fig. 4.1 to improve the segmentation results. In this figure, all of the layers in U-Net are illustrated through blocks visualizing the position of the attention block as well as the skip connections. The network's layer-wise details are already given in Table 4.1 on page no. 51.

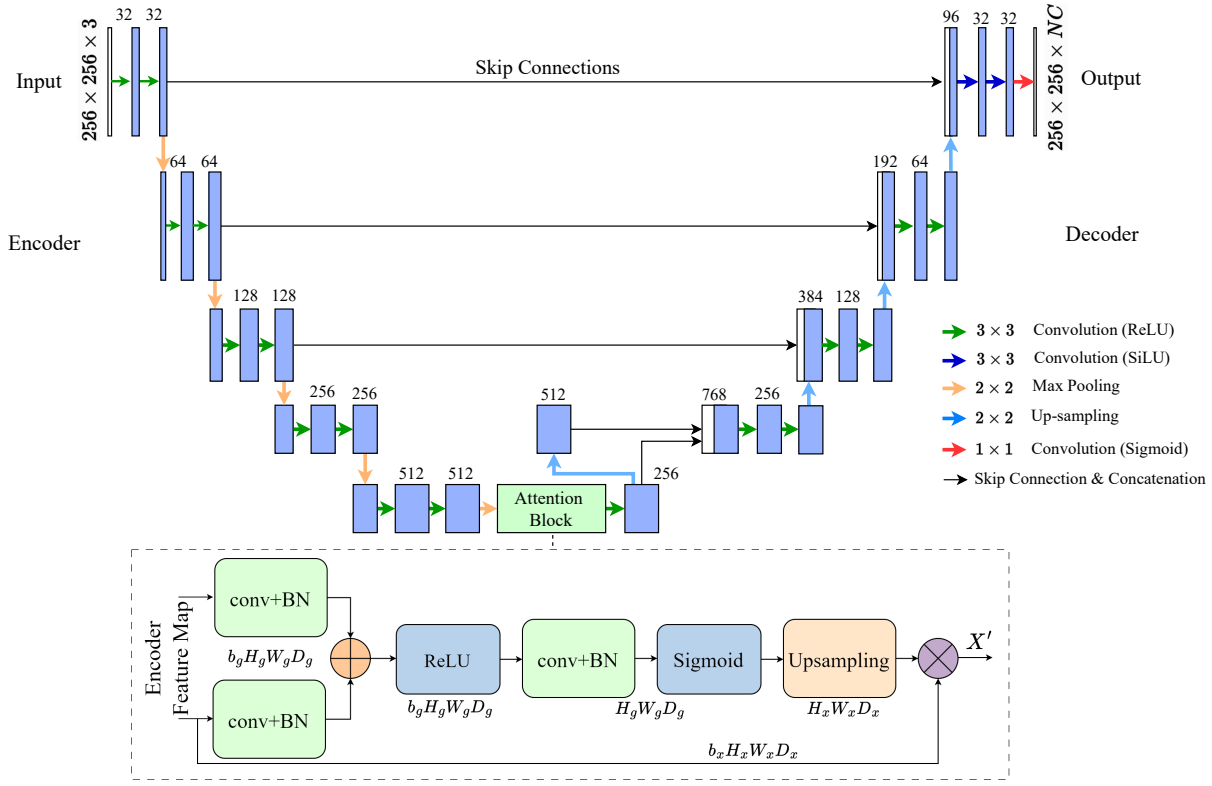


Figure 4.1: The block diagram of the proposed segmentation model with attention block and self-gated activation function.

The Self-gated Activation

The proposed model improves the full-scale (i.e., the same spatial dimension as the input) feature maps' representation capability at the top layers by applying the self-gated activation or the sigmoid linear unit [94]. The rectified linear unit is the standard activation function commonly used in DNNs. The ReLU activation function effectively sets all negative input values to zero

while keeping positive input values unchanged. ReLU is widely used due to its simplicity and efficiency, as it helps mitigate the vanishing gradient problem that can occur with other activation functions like the sigmoid or hyperbolic tangent by allowing gradients to flow more effectively during backpropagation. This results in faster convergence in deep networks. On the other hand, SiLU combines the linear and sigmoid properties, providing a smooth, non-linear transformation that can mitigate the dying ReLU problem. SiLU allows for small negative inputs to contribute to the activation, which can improve the training dynamics and performance of the network. Its differentiable and non-monotonic nature allows for more nuanced activation behaviors compared to ReLU, potentially leading to better learning outcomes in deep learning models.

Meanwhile, ReLU introduces non-linearity to the model and helps to mitigate the vanishing gradient problem. It is defined as

$$\text{ReLU}(x) = \max(0, x), \tag{4.2}$$

where x is the input to the ReLU, representing the weighted sum of inputs in a neural network neuron. The ReLU function returns the input value itself if it is positive, and 0 if it is negative or zero. This simple thresholding operation introduces non-linearity into the neural network while being computationally efficient. On the other hand, the SiLU activation has recently emerged. It is given by

$$\text{SiLU}(x) = x \cdot \sigma(x), \tag{4.3}$$

where x represents the input to the SiLU function, and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid operation applied to x. In addition, the implicit gating mechanism of SiLU can be beneficial for capturing complex relationships in the data. Research indicates that SiLU consistently performs better for computer vision-related tasks than alternative activation functions. This compelling evidence prompted this work to integrate SiLU into the proposed model for improved segmentation results.

4.2.2 Training Strategy

For systematic and comprehensive model development and comparison, this study trains and tests three other configurations besides the proposed model— (i) the baseline (U-Net), (ii) the baseline with full attention + ReLU, and (iii) the baseline with full attention + SiLU. The second (ii) model includes an attention block in every residual connection, and it uses ReLU activation throughout the network, except for the sigmoid used at the output layer. The second experiment focuses on the third (iii) model, which resamples the second one but replaces the ReLU activation function with SiLU in every residual connection. Due to constraints on paper length, the configurations of these two models are not provided here. The `Adam` optimizer [95] with a learning rate of 0.001, and a batch size of 8 is employed to train all models by minimizing the binary cross-entropy objective function, (4.4),

$$E = -\frac{1}{n}\sum_{n=1}^{b}[p_n \log \hat{p}_n + (1-p_n)\log(1-\hat{p}_n)], \tag{4.4}$$

were n represents the number of samples, b is the total number of batches or samples, p_n denotes the

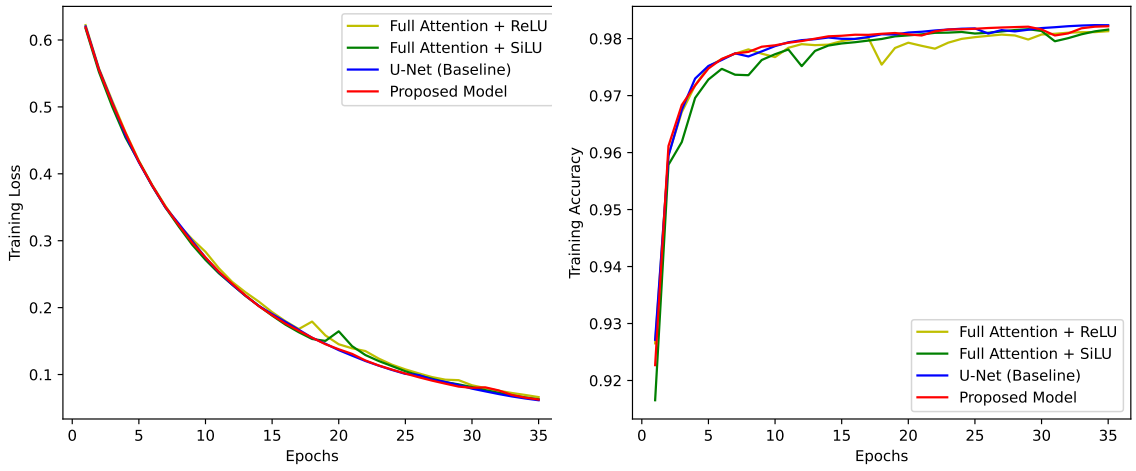


Figure 4.2: Training progress of the four models (cf. Tables 4.2) with respect to accuracy and loss vs. training epochs on the DeepCrack benchmark dataset.

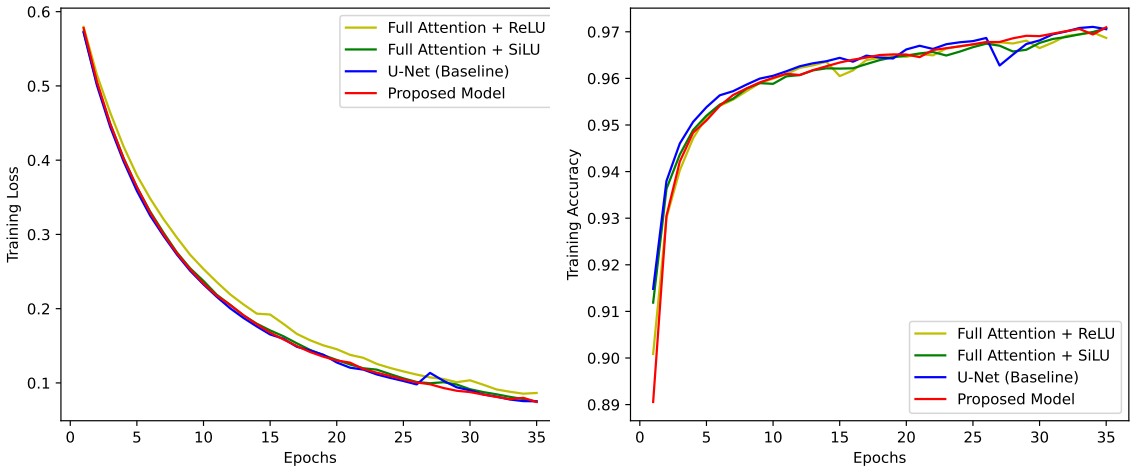


Figure 4.3: Training progress of the four models (cf. Tables 4.2) with respect to accuracy and loss vs. training epochs on the Crack500 benchmark dataset.

Fig. 4.2 and Fig. 4.3 show the overall training progress of all models with respect to loss and accuracy on two individual benchmark datasets, where one can observe that the models reach a plateau at the 35th epoch.

4.3 Experimental Study and Discussion

4.3.1 The Environment

To efficiently train and test the models, substantial storage capacity, and a computational platform are necessary. In this work, all implementations are exclusively developed in `Python 3.10` with the `TensorFlow 2.15.1` deep learning framework on the Google Colab cloud. The training runs on an `NVIDIA Tesla T4` with `15 GB` of `GPU` support. Every experiment follows the same hyper-parameter setting for training and testing for consistency and fair comparison.

4.3.2 Datasets

This study uses the publicly available open-source benchmark dataset–DeepCrack [2] 537 road pavement visuals of size 544×384, capturing various crack types and non-crack conditions. A data preprocessing stage is deployed to resize the samples to a spatial dimension of 256×256 to

match the input layer dimension of the models. To increase the sample size and the diversity of the data, we apply three types of augmentation techniques: 45° rotation, Gaussian blur with sigma 1.5, and brightness with factor 1.5. Thus, the final training and validation sets take 960 and 240 samples, while the test set has 237 samples (a hold-out set exists in the benchmark dataset). An extended experiment is also concluded using another publicly available benchmark dataset–Crack500 [3]. It contains 500 raw samples with the dimension of each image is 640×360. However, the dataset was divided into training, validation, and test sets. However, this study uses 1516 images for training, and 380 samples for validation, while for consistency and fair comparison, it uses the original 200 test samples from the source for testing. The samples are recalled to a spatial dimension of 256×256 to meet the requirement of the proposed model's input layer.

4.3.3 Evaluation Metrics

For a binary segmentation problem, like in this work, the robust evaluation metric used is the mean intersection over union (mIoU) as defined in (4.5). It evaluates the performance of image segmentation models by measuring the accuracy of their predictions across different classes. It provides a comprehensive view of how well a model can segment different parts of an image, making it particularly useful for tasks where distinguishing between various categories is crucial.

$$mIoU = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FP_i + FN_i}, \tag{4.5}$$

where N, TP_i, FP_i, and FN_i stand for the total number of samples, the true positives, false positives, and false negatives for the ith sample. Utilizing these values, other evaluation metrics, such as precision (Pr) and recall (Re), can be calculated, and f1-score (F1) is a measure that combines both precision and recall. Note that a standard threshold value of 0.5 is set for calculating the IoU. It normally takes a value in the range of $[0, 1]$, with 1 indicating a perfect overlap and 0 indicating no overlap between the predicted segmentation and ground truth. Besides, giga-scale floating-point operations (GFLOPs) is also used to measure the models' complexity. GFLOPS is a valuable metric for understanding the computational complexity and efficiency of deep learning

models. By measuring the number of floating-point operations a model can perform per second, it helps in comparing and optimizing models for various applications, especially in scenarios where computational resources are a critical factor. The proposed model's size is 31.96 MB, consumes 226 GFLOPs, and the per-sample inference time is $239 \pm 7\ ms$.

4.3.4 Overall Analysis

Table 4.2 quantitatively analyzes the proposed model's performances as a comparison with other models on the test set of DeepCrack, where models (1), (6), (7), and (8) are fully trained and tested as described in Section 4.3.1, while other models' results are taken from the respective literature.

Table 4.2: Performance comparison of the proposed model with other solutions on the test set of the benchmark dataset—DeepCrack [2], and Crack500 [3]. Note: ↑ and ↓ denote a positive and negative improvement compared to the baseline in mIoU, respectively.

Model	DeepCrack					Crack500				
	Pr	**Re**	**F1**	**mIoU (%)**	**% of Gain**	**Pr**	**Re**	**F1**	**mIoU (%)**	**% of Gain**
1. U-Net (based on [38])	89	87	83.2	78.0	Baseline	70.2	78.0	75.0	58.2	Baseline
2. ECSNet [90]	NA	NA	84.5	73.1	6.41 ↓	NA	NA	NA	NA	NA
3. DMA-Net [96]	86	87	87.0	NA	NA	69.5	80.0	74.4	55.9	3.95 ↓
4. BARNet [97]	NA	NA	NA	NA	NA	66.7	75.7	70.9	53.1	8.76 ↓
5. FFEDN [93]	87	86	86.1	75.7	2.95 ↓	71.0	76.9	73.8	58.6	0.70 ↑
6. Attention U-Net + ReLU	90	85	81.3	75.0	3.85 ↓	62.3	71.0	66.0	55.0	5.50 ↓
7. Attention U-Net + SiLU	90	86	83.0	76.6	1.80 ↓	68.0	77.1	69.6	57.3	1.54 ↓
8. Proposed Model	91	85	87.1	79.0	1.30 ↑	72.6	78.0	76.7	59.6	2.40 ↑

The comparative analysis shows that the proposed model surmounts the other models, including the state-of-the-art solutions, by achieving a mIoU of 79%, which surpasses the baseline by 1.30% mIoU in the pavement crack segmentation results. In addition, the proposed model's performances are evaluated on another test set of Crack500. The ground truth of the test dataset in Crack500 involves some misleading and inaccurate annotations. However, the proposed model gains 2.40% improvement on this dataset in terms of mIoU compared with the baseline model. Where as Table 4.3 represents the comparison of the model's complexity with existing models.

Fig. 4.4 represents the segmentation results of the proposed model on the test set of DeepCrack, along with the baseline and its two variations for qualitative analysis. In this scenario, input images

Table 4.3: Comparison of proposed model complexity with existing models.

Model	Attention	Activation	Trainable Parameters	GFLOPs
1. U-Net (based on [38])	Null	ReLU	7,852,547	223
2. ECSNet [90]	Null	PReLU	410,000	NA
3. DMA-Net [96]	Multi-scale	ReLU	NA	110
4. BAR-Net [97]	All	ReLU	NA	268
5. FFEDN [93]	All	ReLU	NA	525
6. Attention U-Net + ReLU	All	ReLU	8,553,191	235
7. Attention U-Net + SiLU	All	ReLU + SiLU (top)	8,553,191	235
8. Proposed Model	Bottleneck	ReLU + SiLU (top)	8,379,140	226

(their IDs are given for reproducing purposes) are randomly selected from a hold-out set for analysis. Here, five images were selected from 237 test samples to display the most obvious results of

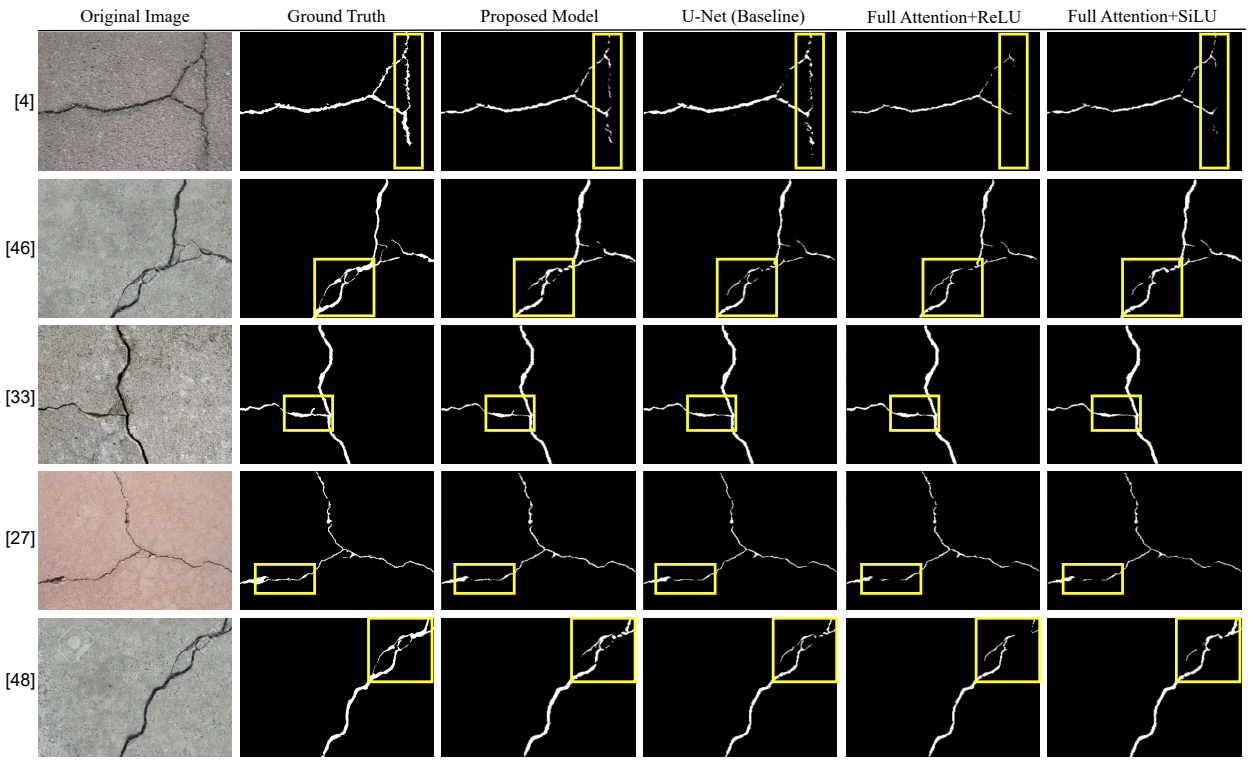


Figure 4.4: Qualitative results of the proposed model compared to three other models on five randomly taken input images from the test set of the DeepCrack dataset.

the model. The first two columns represent the original images and the corresponding ground truth values. The rest of the columns present the experiments. One can observe from the highlighted yellow boxes that the proposed model shows greater robustness in segmenting out the cracks regardless of the fussiness between foreground and background found in the raw inputs. Hence, its

results are much closer to the ground truth, while other models fail to make perfect delineations of the cracks in challenging conditions. Similarly, Fig. 4.5, represents the qualitative results for the Crack500 dataset.

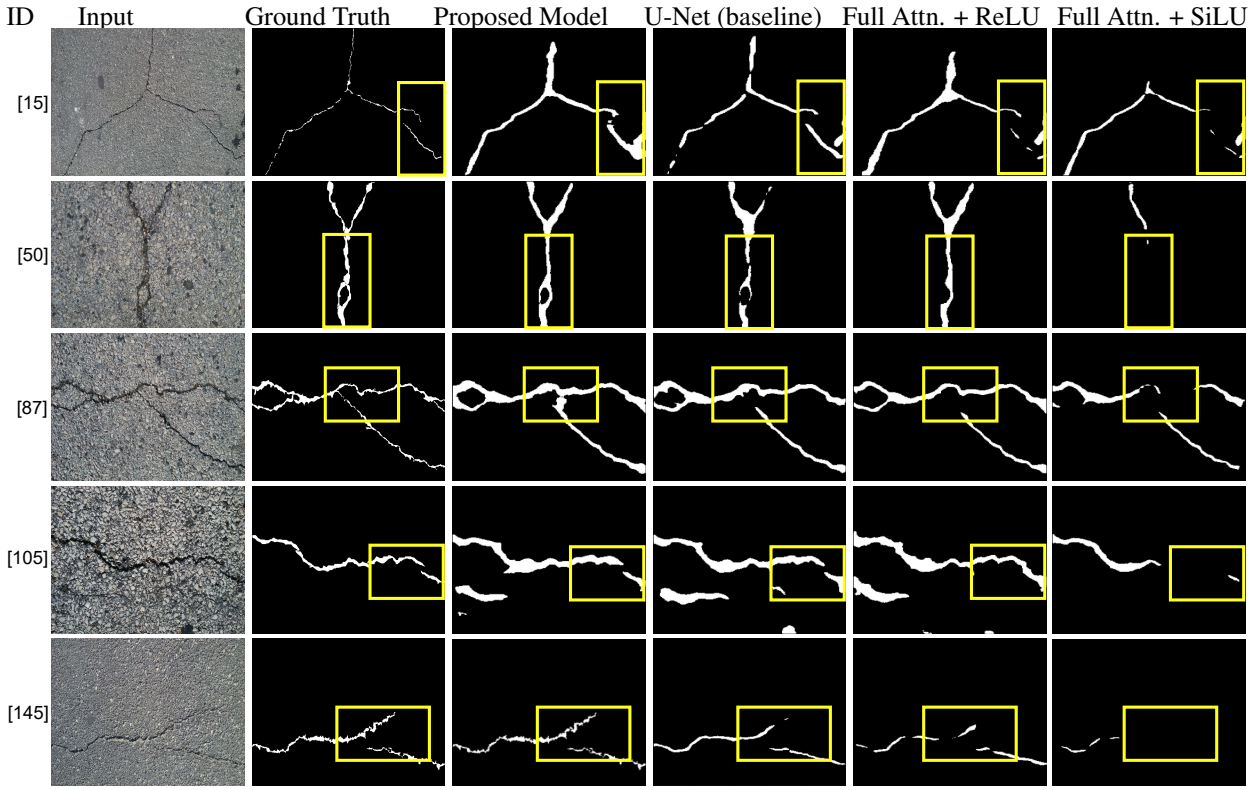


Figure 4.5: Qualitative results of the proposed model compared to three other models on five randomly taken input images from the test set of Crack500 dataset.

In summary, the experimental study reveals that the proposed model effectively balances complexity and performance. Hence, it is observed that incorporating an attention mechanism with SiLU activation enhances the segmentation results. However, it is important to note that excessive use of the attention mechanism may lead to diminished performance.

4.4 Conclusion

As the demand for highway infrastructure usage continues to rise, the deployment of pavement management systems has become increasingly vital to uphold the longevity and reliability of road

pavements. The traditional reliance on manual methods for detecting pavement damages is insufficient to meet the demands. To address this, this work investigates the pavement crack segmentation problem and proposes an improved model. The ablation study on benchmark datasets demonstrates that the proposed model achieves competitive performance compared to cutting-edge methods. The future direction of this work includes the following—(i) It is worth exploring ways to integrate the proposed models with the current practices used for the pavement management systems by various transportation ministries, like the Ministry of Transportation (MTO), to validate their applicability in the practical world; and (ii) Refine the model to work on dynamic video inputs rather than static images, which could involve the integration of model pruning algorithms to streamline the model.

Chapter 5

Improved Semi-supervised Semantic Segmentation

This chapter builds on the insights gained from previous chapters to develop a semi-supervised semantic segmentation model for autonomous driving applications. It explores the use of synthetic data generation and GANs within a semi-supervised training framework to improve the accuracy and effectiveness of semantic segmentation.

5.1 Overview

Semantic segmentation is one of the fundamental problems in the field of computer vision, which involves classifying each pixel of an image into specific semantic categories such as sky, road, car, or person. In traditional supervised learning methods, a large amount of labeled data is required to train models to achieve high accuracy in segmentation tasks. However, obtaining such detailed labels is often a challenging, costly, and time-consuming process, as it requires meticulous human annotation of every pixel in an image. Semi-supervised learning approaches offer a promising solution to this challenge by utilizing both labeled and unlabeled data to enhance segmentation accuracy and reduce the dependency on large labeled datasets.

This research proposes an innovative model that leverages an attention-driven adversarial training strategy within a GAN framework. This method focuses on generating realistic semantic segmentation maps for the unlabeled data while simultaneously improving the segmentation accuracy of the labeled data. The model incorporates an attention mechanism that helps the GAN prioritize significant regions of the image, allowing it to create more accurate segmentation outputs. Furthermore, this study introduces a patch-wise discriminator that is designed to extract rich contextual information from the images, thereby enabling the GAN to produce finer and more coherent segmentation results.

To evaluate the effectiveness of the proposed model, an extensive analysis was conducted using two widely recognized benchmark datasets: Cityscapes and CamVid. These datasets are commonly used in the research community for assessing semantic segmentation models due to their complexity and diversity. The results of the experiments demonstrate that the model achieves state-of-the-art performance in semi-supervised semantic segmentation tasks, significantly surpassing existing methods. By integrating attention mechanisms and patch-wise discriminators within the GAN framework, the proposed approach not only enhances the segmentation accuracy but also contributes to the advancement of semi-supervised learning in the field of semantic segmentation. This research provides a practical solution for improving segmentation accuracy while minimizing the reliance on labeled data, thereby addressing one of the major challenges in computer vision.

5.2 GAN-based Semi-supervised Semantic Segmentation

Image segmentation labels individual pixels providing a deeper understanding of a visual than image-level classification [89]. Semantic segmentation, in particular, has garnered significant interest across sectors, such as agriculture, healthcare, transportation, and infrastructure management [17]. For example, in a street scene, semantic segmentation identifies each pixel as a road, sidewalk, car, pedestrian, building, sky, or tree. For autonomous driving, this information is crucial for accurate decision-making [98, 99].

Building segmentation models, such as U-Net– a convolutional neural network using supervised learning, is labor-intensive and time-consuming due to the need for labeled data, and often impractical for certain applications, like medicine [32]. Consequently, there has been an interest in semi-supervised learning approaches, which try to use publicly available unlabeled data to enhance the models' performance. Comparable or even improved segmentation results can be achieved from this. Meanwhile, integrating semi-supervised learning techniques with GAN has emerged as a compelling method for semantic segmentation. GANs are renowned for their ability to generate realistic data samples and their learning scheme, which leverages unlabeled data effectively. GANs consist of a discriminator network that distinguishes between real and synthesized data and a generator network to produce semantically meaningful segmentation maps. The integration of semi-supervised learning in GAN can revolutionize semantic segmentation in various domains, including medical image analysis, autonomous driving, and object detection [100–102]. Thus, this work aims to delve into these challenges and opportunities, offering insights into cutting-edge methodologies and assessing their performance across diverse datasets and application domains. It also provides a roadmap for future research avenues in the realm of semi-supervised semantic segmentation utilizing GAN models. Performance investigation on two benchmark datasets for semantic segmentation reveals the effectiveness of the suggested methodology in comparison with the state-of-the-art method. The main contributions of this work are as follows.

- Introducing an attention mechanism and self-gated activation in the U-Net architecture, which is the generator of the proposed patchGAN framework.
- The model training follows a semi-supervised strategy.
- Demonstrating the model's effectiveness on two benchmark datasets, namely Cityscapes and CamVid.

5.3 Methodology

Fig. 5.1 depicts the proposed GAN-based image segmentation network comprising a generator and a discriminator networks, G and D, respectively. The generator is an encoder-decoder ar-

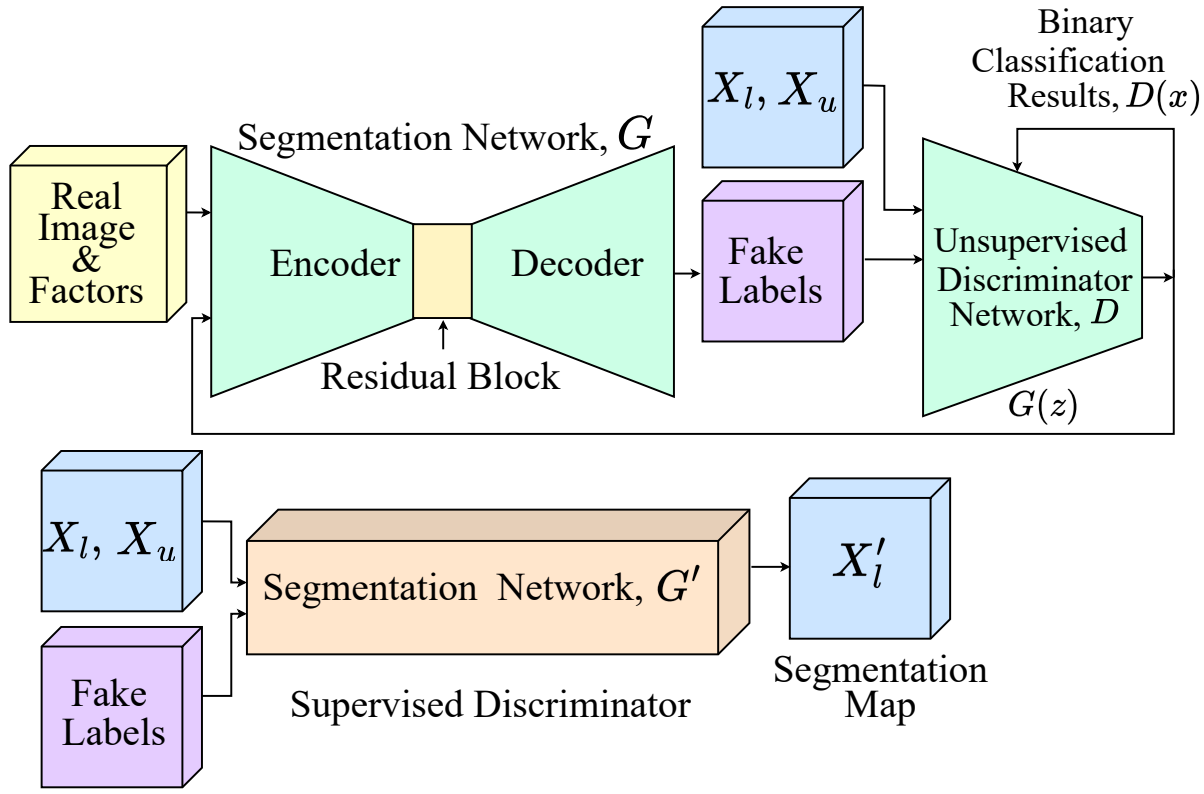


Figure 5.1: Overview of the proposed semi-supervised GAN network. A U-net-based model is used as the generator G. The discriminator D is used for $N \times N$ patch pixel-level segmentation.

chitecture and is a common choice for image segmentation. The input image is compressed by the encoder into a lower-dimensional representation, and the decoder then reconstructs the input from that representation. The discriminator network functions as a scene parser with a patch-out module. It receives the generated samples from G, unlabeled data $\mathbf{X}_u$, and pixel-level annotation $\mathbf{X}_l$. It outputs a binary classification $D(x)$—`real` or `fake`, for each $N \times N$ patch in the input image. In a semi-supervised adversarial training framework, the discriminator reduces the likelihood of synthetically generated images being classified as real. The generator produces images, and the discriminator classifies them as real, and a secondary network can be used to promote high confidence in semantic labels for real images. This framework integrates additional semantic

knowledge into the adversarial learning process to enhance the generator's performance through combined analysis.

5.3.1 The Generator Subnetwork

The generator subnetwork is the segmentation model. In this work, the proposed model in Chapter 4 (cf. Table 4.1) is repurposed as the generator. The input layer of the model is configured to accommodate visible images of dimensions 256×256. Consequently, both the encoding and decoding paths employ standard convolution operations with a 3×3 kernel size, a stride rate of 1, and padding set to `'same'`, followed by leaky rectified linear unit (LeakyReLU) activation. The LeakyReLU defined as in (5.1) is a computationally efficient non-linear activation that allows the model to capture complex patterns while mitigating the vanishing gradient problem.

$$\text{LeakyReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ \alpha x & \text{otherwise} \end{cases}, \tag{5.1}$$

where x represents the input to the LeakyReLU, typically the weighted sum of inputs in a neural network neuron, and α is a small positive constant determining the function's slope for negative inputs. LeakyReLU, in contrast to the conventional ReLU function, allows a small, non-zero gradient when the input is negative, which might be advantageous for detecting complex generalized patterns. To down-sample spatial dimensions and extract crucial features, the encoding path incorporates max-pooling layers with a stride of 2. Conversely, the decoding sub-network utilizes interpolation-based upsampling (`nearest-neighbor`) to regain the spatial dimensions. However, the bottleneck and top layers of the proposed model are specifically tailored to capture valuable semantic information better. Introducing an attention block (`L15 - L23`) between the encoding and decoding sub-networks enhances learned features. Consequently, the top layer in the proposed architecture employs the SiLU activation in (5.2), contributing to improved generaliza-

tion performance. The SiLU activation function is defined as follows:

$$\text{SiLU}(x) = x \cdot \sigma(x), \tag{5.2}$$

where x represents the input to the SiLU function, and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the sigmoid operation applied to x. Unlike ReLU, which outputs zero for negative inputs, SiLU preserves some information from negative inputs. Finally, the output layer of the architecture employs a convolution operation with a 1×1 kernel, resulting in a semantic segmentation map predicted by a `tanh` activation in (5.3) that ensures useful output range and facilitates stable training.

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}. \tag{5.3}$$

The attention mechanism, as in the previous Chapter (cf. Fig. 4.1 of Chapter 4), enables the model to dynamically adjust the importance of different spatial locations in the input image. By incorporating attention mechanisms into the network architecture, the model can selectively attend to informative regions while filtering out distractions, leading to a more accurate and detailed segmentation map. It involves feature concatenation and element-wise addition to produce an information-rich feature map. Specifically, the sigmoid function, σ generates a gated weight; after that, it is multiplied by the input feature, X to yield the refined attention feature map, X', as defined by (5.4).

$$X^{'} = (X \odot \sigma(f_g \oplus f_x)), \tag{5.4}$$

where f_g ($[b_g H_g W_g D_g]$) and f_x ($[b_x H_x W_x D_x]$) represent the feature maps obtained from upsampling and encoder operations, respectively, followed by element-wise addition. The subsequent application of the σ operation generates the final feature map using dot-product operation. The attention block enhances the model's adaptability by dynamically adjusting the relevance of different spatial information, resulting in more precise and context-aware segmentation outcomes.

Table 5.1: Architectural detail of the patch-wise discriminator

Layer ID	Layer type $A(k, s)$	Output Shape $[b, H, W, D]$	Input
Input	Input Layer	$[b, 256, 256, 6]$	mini-batch
L1	Conv (4,4)→ LeakyReLU	$[b, 128, 128, 64]$	Input
L2	Conv (4,4)→ BN+LeakyReLU	$[b, 64, 64, 128]$	L1
L3	Conv (4,4)→ BN+LeakyReLU	$[b, 32, 32, 256]$	L2
L4	Conv (4,4)→ BN+LeakyReLU	$[b, 16, 16, 512]$	L3
L5	Conv (4,4)→ BN+LeakyReLU	$[b, 16, 16, 1]$	L4

5.3.2 Discriminator Subnetwork

The discriminator network is formulated to differentiate between authentic and synthetically generated images. It receives two input images, the source and target images (segmented image). These images are concatenated channel-wise and fed into the network depicted in Table 5.1. The network consists of five convolutional (ConV) layers, each employing a 4×4 kernel with a stride of 2×2, except for the last layer, which uses a stride of 1×1. The learned features of each Conv layer are passed through a `LeakyReLU` activation with a slope of 0.2. Batch normalization (BN) is applied after each ConV operation to stabilize training. Eventually, two final outputs are obtained by applying a `sigmoid` activation with a 16×16 patch size for determining the path's authenticity of (`real` or `fake`) using discriminator loss $D(x)$. To the next step, apply a `softmax` activation with `sparse_categorical_crossentropy (scce)` loss, $G(z)$ in (5.5) for predicting segmentation maps according to respective categories. Thus, the discriminator is trained using the total loss $\mathcal{L}_{GAN}(G, D)$ as defined in (2.1).

$$G(z) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c}\log(p_{i,c}), \tag{5.5}$$

here the sparse categorical cross-entropy directly uses the true class label y_i for each pixel i among the c number of classes.

5.3.3 Training Details

The stochastic gradient (SGD) optimizer is used to train the proposed segmentation subnetwork with a momentum of 0.9, a weight decay of 0.0005, and an initial learning rate of 0.0002 with a

polynomial learning rate schedule. The discriminator subnetwork is optimized using the Adam optimizer with a base learning rate of 0.0001 and the exponential decay rates of the moving averages of the gradient (β_1) and the squared gradient (β_2) are set to 0.9, and 0.99, respectively. Except for the batch size, set to 2 for the CamVid dataset and 5 for the Cityscapes dataset, these hyperparameters are the same for all experiments. The mean Intersection over Union (mIoU), specified in (5.6), is used to evaluate the model's performance.

$$mIoU = \frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FP_i + FN_i}, \tag{5.6}$$

where N, TP_i, FP_i, and FN_i stand for the total number of samples, the true positives, false positives, and false negatives for the ith sample. Additionally, giga-scale floating-point operations (GFLOPs) determine model complexity. A more significant GFLOP indicates more computing needs. The proposed model has a generator and discriminator with 31.96 MB and 10.67 MB, respectively, and GFLOPs is 12.1.

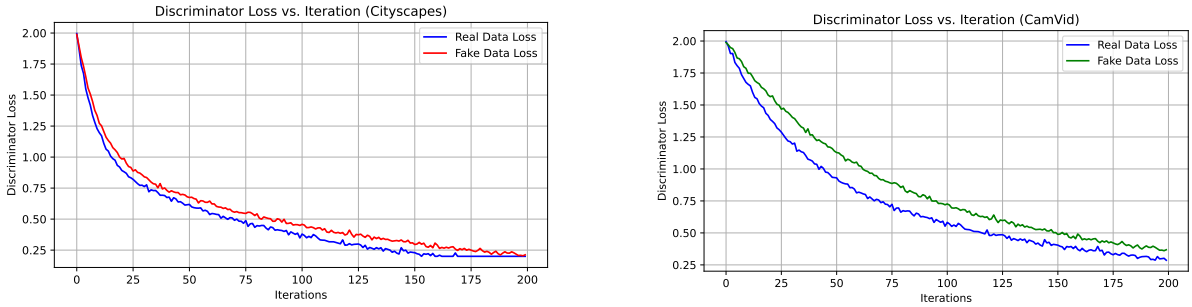


Figure 5.2: Training progress of two benchmark datasets: Cityscapes, and CamVid.

Fig. 5.2 illustrates the training progress of the discriminator to generate fake images (generated). A total of 200 iterations were performed to generate images according to real images (ground truth). This shows the effect of discriminator loss, which is useful to understand the performance of generating fake images. For generator training, these images will be considered as unlabeled data along with the labeled data.

5.4 Experimental Analysis

5.4.1 Environment Setup

All implementations are completed on Google's CoLab notebook with Python 3 and other libraries, such as Keras, TensorFlow, and OpenCV. A substantial amount of RAM or GPU support is necessary to train the proposed model; in this case, we use a Tesla T4 12GB GPU. Finally, the proposed model was tested on the Jupyter Notebook with 32GB GPU support on a local PC.

5.4.2 Datasets

The $Cityscapes$ [103] dataset consists of 50 recordings of driving scenarios from which 2975, 500, and 1525 images are extracted and labeled with 19 classes for training, validation, and testing, respectively. Each annotated frame is the $20th$ frame in a $30th$ frame snippet, and the training process considers these images with annotations. We resized the provided image from 1024×2048 to 256×256 without arbitrary cropping or scaling. To evaluate the model, images are collected from the val set of Cityscapes.

The $CamVid$ [104] has almost 10 minutes of recordings, encompassing over 11K frames, of which 701 images with a resolution of 960×720 are pixel-level annotated. There are thirty-two semantic labels. In this thesis, we used 468 samples with a dimension of 256×256 as a training dataset for fully supervised learning and different ratios of unlabeled frames for semi-supervised learning and evaluated the model with the $test$ set of 233 samples.

5.4.3 Overall Discussion

This thesis randomly interleaves labeled and unlabeled data for semi-supervised training and jointly updates the generator and discriminator sub-networks. In each iteration, only the batch containing the ground truth data is used to train the discriminator. The experiments are repeated several times with different random samples to ensure the robustness of the model. During training, the total training dataset holds labeled samples according to the following fractions: 1/30, 1/8, 1/4,

Table 5.2: Quantitative analysis of various semi-supervised semantic segmentation methods on mIoU in % for various ratios of labeled and unlabeled data (1/30, 1/8/,1/4) used in training.

Method	1/30	1/8	1/4	Fully Labeled
Cityscapes Validation Dataset				
Hung *et al.* [67]	NA	58.8	62.3	NA
s4GAN [73]	NA	59.3	61.9	65.8
C3-SemiSeg [71]	55.1	63.2	65.5	69.5
KE-GAN [72]	NA	66.9	70.6	75.3
This work	**61.4**	**67.8**	**76.3**	**81.9**
CamVid Test Dataset				
MSCFNet *et al.* [58]	NA	NA	NA	69.3
This work	**59.8**	**65.9**	**73.5**	**77.6**

NA - Result is not found in the literature. The best results are in boldface.

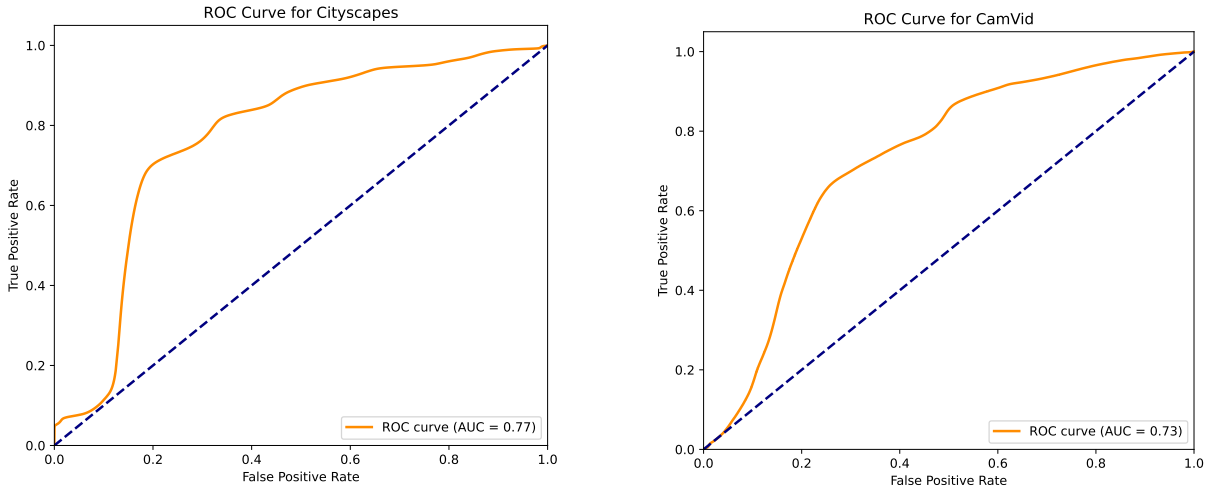


Figure 5.3: AUC-ROC of the proposed semi-supervised model on Cityscapes and CamVid datasets.

and fully labeled (supervised) as indicated in Table 5.2. Compared to the best existing model–KE-GAN [72], the proposed model on the Cityscapes dataset achieves 1.35% and 8.92% improvement for the 1/8 and 1/4 data splits, respectively. On the other hand, the model gets 11.97% improvement for a fully supervised CamVid dataset compared to MSCFNet [58]. Furthermore, achieving a 75% and 73% AUC in Fig. 5.3 suggests that the model is able to capture some of the underlying patterns in the data, although it may struggle with certain classes or in more challenging scenarios, such as heavily occluded objects or rare classes. Fig. 5.4 on page no. 72 and Fig. 5.5 on page no. 73 present a few qualitative results on Cityscapes and CamVid datasets, respectively. Note that the outputs

are rescaled to maintain the aspect ratio of the original inputs. The samples' indexes in the actual datasets are indicated on the left side of the image for better understanding and reproducibility. A detailed observation proves that segmentation maps are close to the ground truth counterparts. Images are randomly selected from both datasets to generate the predicted ground truth. In the end, the proposed U-net as a generator network with a patch-wise discriminator works better for semantic segmentation.

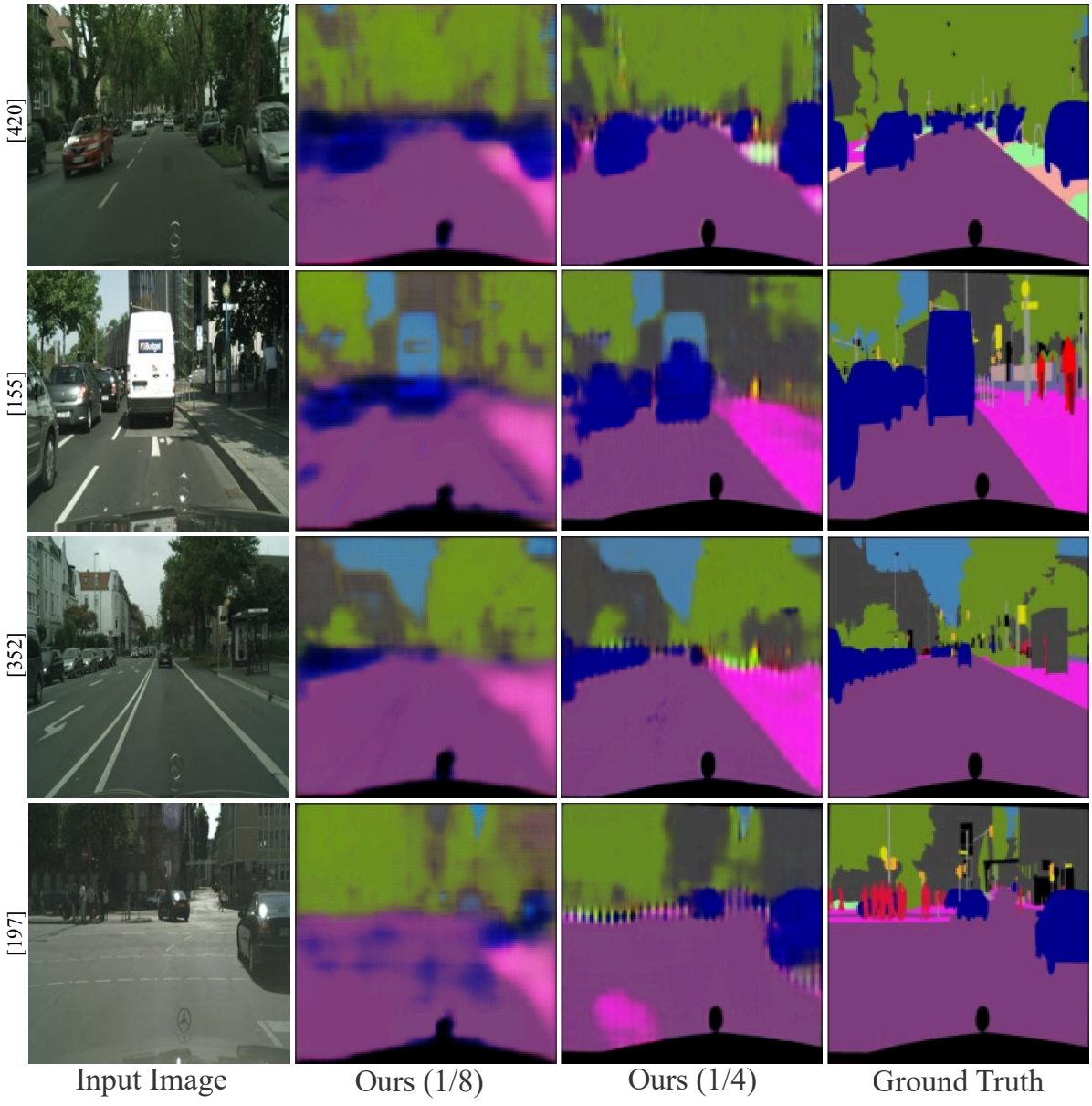


Figure 5.4: Qualitative results of the proposed semi-supervised approach for four randomly selected images from the Cityscapes validation dataset.

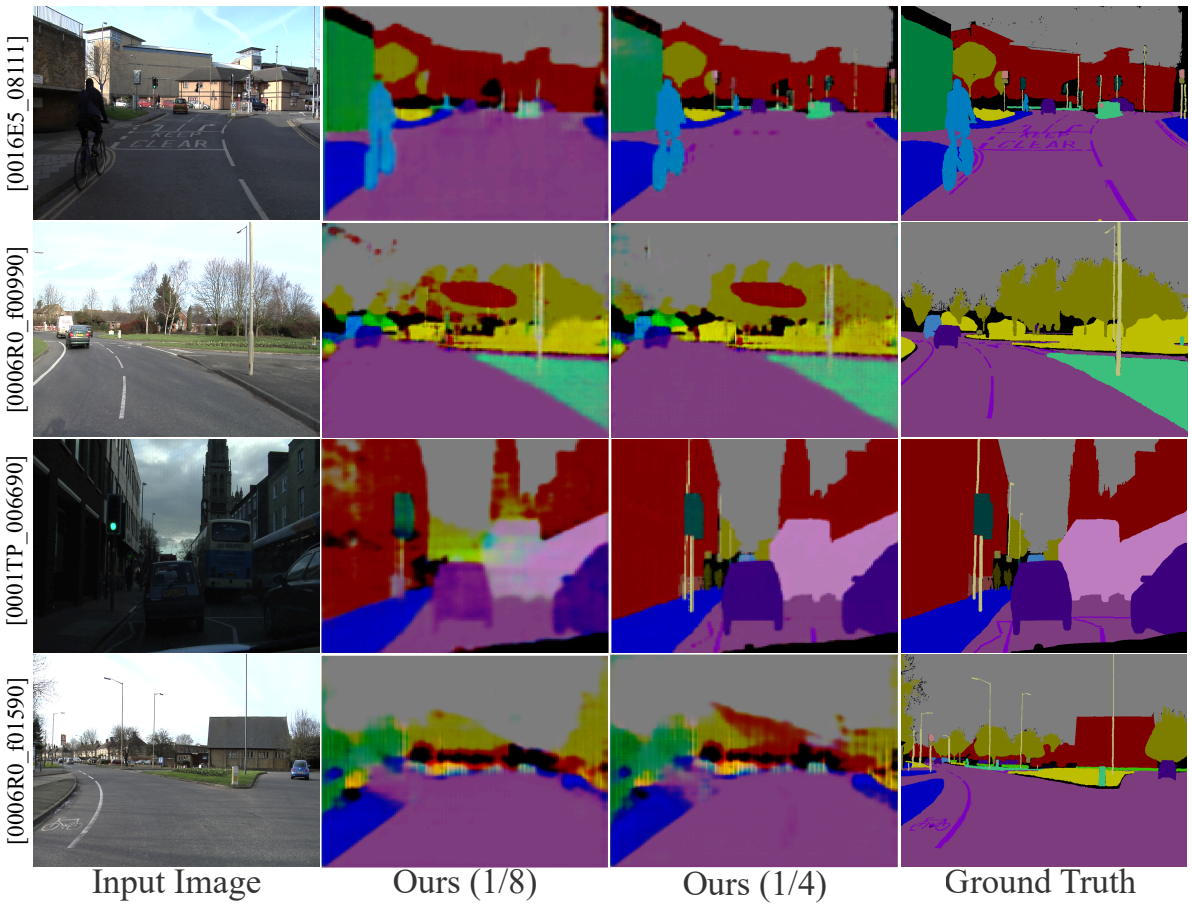


Figure 5.5: Qualitative results of the proposed semi-supervised approach for four randomly selected images from the CamVid test dataset.

5.5 Conclusion

This work overcomes the attention inefficiencies in existing semi-supervised approaches for image semantic segmentation. The proposed discriminator and generator networks coordinate the updating of the segmentation model. A patch-wise discriminator extracts more contextual information from the input scenes. The extensive experiments on two benchmark datasets prove the effectiveness of the proposed model. In the future, two key areas will be focused for further investigation: (i) enhancing the capacity of generator networks to express latent geometric properties, such as creating a high-quality non-Euclidean feature space, and (ii) developing a few-shot semantic seg-

mentation technique, which is essential for domains like remote sensing where sample collection is prohibitively expensive.

Chapter 6

Concluding Insights and Future Directions

This thesis demonstrates the significant potential of the proposed semi-supervised image segmentation and classification framework using deep convolutional neural networks. By effectively leveraging both labeled and unlabeled data, the framework addresses critical challenges in traditional supervised learning methods, such as data scarcity and high labeling costs. The integration of adversarial learning and pseudo-labeling techniques has notably enhanced the model's robustness and accuracy, achieving state-of-the-art performance on benchmark datasets.

Despite these advancements, there are several limitations inherent to the current approach. One key limitation is the dependency on the datasets, may not fully represent the diversity of real-world scenarios. The dataset's specific domain could restrict the generalizability of the model to other types of images or environments. Additionally, the limited computational resources available during this research constrained the ability to experiment with more complex models or perform extensive hyperparameter tuning. This limitation could impact the overall efficiency and performance of the framework.

Moreover, the effectiveness of the semi-supervised learning approach hinges on finding the optimal balance between labeled and unlabeled data. An inappropriate balance can lead to suboptimal performance, with risks of overfitting or inadequate learning from unlabeled data. Over-reliance on synthetic data in GAN-based approaches and difficulties in generalizing across diverse domains and image types pose further constraints.

Future research should focus on expanding the scalability of the framework to larger and more varied datasets, integrating domain adaptation techniques to enhance generalization, and developing more accurate pseudo-labeling methods. Exploring alternative neural network architectures and novel training strategies could further improve efficiency and accuracy in semi-supervised image segmentation and classification. Addressing these research directions will advance the field of semi-supervised learning in computer vision, leading to more effective and reliable image analysis systems.

Ultimately, the proposed methods promise to enhance the efficiency and performance of image segmentation and classification applications, contributing to robust solutions in areas such as medical imaging, autonomous driving, and remote sensing.

Bibliography

[1] S. N. Ali, M. T. Ahmed, J. Paul, T. Jahan, S. Sani, N. Noor, and T. Hasan, "Monkeypox skin lesion detection using deep learning models: A feasibility study," *arXiv preprint arXiv:2207.03342*, 2022.

[2] Y. Liu, J. Yao, X. Lu, R. Xie, and L. Li, "Deepcrack: A deep hierarchical feature learning architecture for crack segmentation," *Neurocomputing*, vol. 338, pp. 139–153, 2019.

[3] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *2016 IEEE Intl. Conf. on image processing (ICIP)*. IEEE, 2016, pp. 3708–3712.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[5] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/papers/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.pdf

[6] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, 2019, pp. 6105–6114. [Online]. Available: http://proceedings.mlr.press/v97/tan19a/tan19a.pdf

[7] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár, "Panoptic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9404–9413.

[8] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2359–2367.

[9] R. Sharma, M. Saqib, C.-T. Lin, and M. Blumenstein, "A survey on object instance segmentation," *SN Computer Science*, vol. 3, no. 6, p. 499, 2022.

[10] Q. Yang, J. Peng, and D. Chen, "A review of research on instance segmentation based on deep learning," in *International Conf. on Computer Engineering and Networks*. Springer, 2023, pp. 43–53.

[11] X. Ma, X. Zhang, M.-O. Pun, and M. Liu, "A multilevel multimodal fusion transformer for remote sensing semantic segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–15, 2024.

[12] Q. Zhou, L. Wang, G. Gao, B. Kang, W. Ou, and H. Lu, "Boundary-guided lightweight semantic segmentation with multi-scale semantic context," *IEEE Trans. on Multimedia*, vol. 26, pp. 7887–7900, 2024.

[13] Y. Yin, H. Chen, W. Zhou, J. Deng, H. Xu, and H. Li, "Revisiting open-set panoptic segmentation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, 2024, pp. 6747–6754.

[14] F. Hong, L. Kong, H. Zhou, X. Zhu, H. Li, and Z. Liu, "Unified 3d and 4d panoptic segmentation via dynamic shifting networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3480–3495, 2024.

[15] J. Hu, L. Huang, T. Ren, S. Zhang, R. Ji, and L. Cao, "You only segment once: Towards real-time panoptic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 819–17 829.

[16] X. Li and D. Chen, "A survey on deep learning-based panoptic segmentation," *Digital Signal Processing*, vol. 120, p. 103283, 2022.

[17] V. Marsocci, S. Scardapane, and N. Komodakis, "Mare: Self-supervised multi-attention resu-net for semantic segmentation in remote sensing," *Remote Sensing*, vol. 13, no. 16, p. 3275, 2021.

[18] M. Soliman, C. Lehman, and G. AlRegib, "S 6: semi-supervised self-supervised semantic segmentation," in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 1861–1865.

[19] K. Zhu, W. Zhai, Z.-J. Zha, and Y. Cao, "Self-supervised tuning for few-shot segmentation," 2020.

[20] B. Kayalibay, G. Jensen, and P. van der Smagt, "Cnn-based segmentation of medical imaging data," 2017.

[21] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[22] A. K. Chaudhary, P. K. Gyawali, L. Wang, and J. B. Pelz, "Semi-supervised learning for eye image segmentation," in *ACM Symposium on Eye Tracking Research and Applications*, 2021, pp. 1–7.

[23] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," in *Advances in Neural Informa-*

tion Processing Systems, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014.

[24] X. Zhang, W. Xie, C. Huang, Y. Wang, Y. Zhang, X. Chen, and Q. Tian, "Self-supervised tumor segmentation through layer decomposition," *arXiv preprint arXiv:2109.03230*, 2021.

[25] Z. Wang, K. Wang, Z. Liu, X. Wang, and S. Pan, "A cognitive vision method for insect pest image segmentation," *IFAC-PapersOnLine*, vol. 51, no. 17, pp. 85–89, 2018.

[26] R. Güldenring and L. Nalpantidis, "Self-supervised contrastive learning on agricultural images," *Computers and Electronics in Agriculture*, vol. 191, p. 106510, 2021.

[27] J. Novosel, P. Viswanath, and B. Arsenali, "Boosting semantic segmentation with multi-task self-supervised learning for autonomous driving applications," in *Proc. of NeurIPS-Workshops*, vol. 3, 2019.

[28] A. Kalapos and B. Gyires-Tóth, "Self-supervised pretraining for 2d medical image segmentation," in *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*. Springer, 2023, pp. 472–484.

[29] W. Boerdijk, M. Sundermeyer, M. Durner, and R. Triebel, "Self-supervised object-in-gripper segmentation from robotic motions," *arXiv preprint arXiv:2002.04487*, 2020.

[30] A. Valada, R. Mohan, and W. Burgard, "Self-supervised model adaptation for multimodal semantic segmentation," *International Journal of Computer Vision*, vol. 128, no. 5, pp. 1239–1285, 2020.

[31] D. Mahapatra, A. Poellinger, L. Shao, and M. Reyes, "Interpretability-driven sample selection using self supervised learning for disease classification and segmentation," *IEEE transactions on medical imaging*, vol. 40, no. 10, pp. 2548–2562, 2021.

[32] N. Jahan, G. Bajwa, and T. Akilan, "Federated learning-assisted self-supervised cnn for monkeypox diagnosis," in *2023 IEEE Western New York Image and Signal Processing Workshop*, 2023, pp. 1–5.

[33] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[34] X. Zhu and A. B. Goldberg, *Introduction to semi-supervised learning*. Springer Nature, 2022.

[35] A. Agrawala, "Learning with a probabilistic teacher," *IEEE Transactions on Information Theory*, vol. 16, no. 4, pp. 373–379, 1970.

[36] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[37] A. Odena, "Semi-supervised learning with generative adversarial networks," *arXiv preprint arXiv:1606.01583*, 2016.

[38] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th Intl. Conf., Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105. [Online]. Available: https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[41] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*,

2015, pp. 3431–3440. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Long_Fully_Convolutional_Networks_2015_CVPR_paper.pdf

[42] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[43] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[44] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969. [Online]. Available: https://openaccess.thecvf.com/content_iccv_2017/papers/He_Mask_R-CNN_ICCV_2017_paper.pdf

[45] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.

[46] H. Hu, Y. Wu, Y. Liu, J. Yan, W. Shen, X. Yu, and Y. Xiong, "Pyramid scene parsing network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2881–2890. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/papers/Zhao_Pyramid_Scene_Parsing_CVPR_2017_paper.pdf

[47] T. Akilan, Q. J. Wu, and H. Zhang, "Effect of fusing features from multiple dcnn architectures in image classification," *IET Image Processing*, vol. 12, no. 7, pp. 1102–1110, 2018.

[48] R. Pramanik, B. Banerjee, G. Efimenko, D. Kaplun, and R. Sarkar, "Monkeypox detection from skin lesion images using an amalgamation of cnn models aided with beta function-based normalization scheme," *Plos one*, vol. 18, no. 4, p. e0281815, 2023.

[49] C. Sitaula and T. B. Shahi, "Monkeypox virus detection using pre-trained deep learning-based approaches," *Journal of Medical Systems*, vol. 46, no. 11, p. 78, 2022.

[50] W. Liu, "Implementation of detection of skin lesions in monkeypox based on a deep learning model: using an improved bilinear pooling model," in *Second International Conference on Biological Engineering and Medical Science*, vol. 12611. SPIE, 2023, pp. 212–219.

[51] V. Alcalá-Rmz, K. E. Villagrana-Bañuelos, J. M. Celaya-Padilla, J. I. Galván-Tejada, H. Gamboa-Rosales, and C. E. Galván-Tejada, "Convolutional neural network for monkeypox detection," in *International conference on ubiquitous computing and ambient intelligence*. Springer, 2022, pp. 89–100.

[52] T. Morita and X.-H. Han, "Investigating self-supervised learning for skin lesion classification," in *2023 18th International Conference on Machine Vision and Applications (MVA)*. IEEE, 2023, pp. 1–5.

[53] D. Chen, Y. Chen, Y. Li, F. Mao, Y. He, and H. Xue, "Self-supervised learning for few-shot image classification," in *International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2021, pp. 1745–1749.

[54] S.-C. Huang, A. Pareek, M. Jensen, M. P. Lungren, S. Yeung, and A. S. Chaudhari, "Self-supervised learning for medical image classification: a systematic review and implementation guidelines," *NPJ Digital Medicine*, vol. 6, no. 1, p. 74, 2023.

[55] A. Chebli, A. Djebbar, and H. F. Marouani, "Semi-supervised learning for medical application: A survey," in *2018 International Conference on Applied Smart Systems (ICASS)*, 2018, pp. 1–9.

[56] M. N. Hossen, V. Panneerselvam, D. Koundal, K. Ahmed, F. M. Bui, and S. M. Ibrahim, "Federated machine learning for detection of skin diseases and enhancement of internet of medical things (iomt) security," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 2, pp. 835–841, 2023.

[57] J. Wicaksana, Z. Yan, X. Yang, Y. Liu, L. Fan, and K.-T. Cheng, "Customized federated learning for multi-source decentralized medical image classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 11, pp. 5596–5607, 2022.

[58] G. Gao, G. Xu, Y. Yu, J. Xie, J. Yang, and D. Yue, "Mscfnet: A lightweight network with multi-scale context fusion for real-time semantic segmentation," *IEEE Trans. on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25 489–25 499, 2021.

[59] Z. Wang, J. Zhang, Z. Liu, S. Chen, and D. Lu, "An improved u-net network for medical image segmentation," in *2023 IEEE 10th International Conf. on Cyber Security and Cloud Computing (CSCloud)/2023 IEEE 9th International Conf. on Edge Computing and Scalable Cloud (EdgeCom)*, 2023, pp. 292–297.

[60] S. Fang, B. Zhang, and J. Hu, "Improved mask r-cnn multi-target detection and segmentation for autonomous driving in complex scenes," *Sensors*, vol. 23, no. 8, p. 3853, 2023.

[61] K. Chowdhary and K. Chowdhary, "Natural language processing," *Fundamentals of artificial intelligence*, pp. 603–649, 2020.

[62] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proceedings of the IEEE/CVF Conf. on computer vision and pattern recognition*, 2019, pp. 3146–3154.

[63] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," in *Proceedings of the IEEE/CVF Conf. on computer vision and pattern recognition*, 2020, pp. 11 534–11 542.

[64] B. Lei, Z. Xia, F. Jiang, X. Jiang, Z. Ge, Y. Xu, J. Qin, S. Chen, T. Wang, and S. Wang, "Skin lesion segmentation via generative adversarial networks with dual discriminators," *Medical Image Analysis*, vol. 64, p. 101716, 2020.

[65] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[66] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," in *NIPS Workshop on Adversarial Training*, 2016.

[67] W. C. Hung, Y. H. Tsai, Y. T. Liou, Y.-Y. Lin, and M. H. Yang, "Adversarial learning for semi-supervised semantic segmentation," in *29th British Machine Vision Conf., BMVC 2018*, 2018.

[68] D. Li, J. Yang, K. Kreis, A. Torralba, and S. Fidler, "Semantic segmentation with generative models: Semi-supervised learning and strong out-of-domain generalization," in *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2021, pp. 8300–8311.

[69] S. Hong, H. Noh, and B. Han, "Decoupled deep neural network for semi-supervised semantic segmentation," *Advances in neural information processing systems*, vol. 28, 2015.

[70] O. T. Nartey, G. Yang, J. Wu, and S. K. Asare, "Semi-supervised learning for fine-grained classification with self-training," *IEEE Access*, vol. 8, pp. 2109–2121, 2019.

[71] Y. Zhou, H. Xu, W. Zhang, B. Gao, and P.-A. Heng, "C3-semiseg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing," in *Proceedings of the IEEE/CVF International Conf. on Computer Vision*, 2021, pp. 7036–7045.

[72] M. Qi, Y. Wang, J. Qin, and A. Li, "Ke-gan: Knowledge embedded generative adversarial networks for semi-supervised scene parsing," in *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2019, pp. 5237–5246.

[73] S. Mittal, M. Tatarchenko, and T. Brox, "Semi-supervised semantic segmentation with high- and low-level consistency," *IEEE Trans. on Pattern Analy. and Mach. Intellige.*, vol. 43, no. 4, pp. 1369–1379, 2019.

[74] C. for Disease Control, Prevention *et al.*, "Monkeypox and orthopoxvirus outbreak global map," 2022.

[75] S. Chuprov, A. N. Satam, and L. Reznik, "Are ml image classifiers robust to medical image quality degradation?" in *2022 IEEE Western New York Image and Signal Processing Workshop*, 2022, pp. 1–4.

[76] Z. Jezek, A. Gromyko, and M. Szczeniowski, "Human monkeypox." *Journal of Hygiene, Epidemiology, Microbiology, and Immunology*, vol. 27, no. 1, pp. 13–28, 1983.

[77] A. M. McCollum and I. K. Damon, "Human monkeypox," *Clinical infectious diseases*, vol. 58, no. 2, pp. 260–267, 2014.

[78] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*. PMLR, 2017, pp. 1273–1282.

[79] J. Luo and S. Wu, "Fedsld: Federated learning with shared label distribution for medical image classification," in *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2022, pp. 1–5.

[80] J. Wicaksana, Z. Yan, D. Zhang, X. Huang, H. Wu, X. Yang, and K.-T. Cheng, "Fedmix: Mixed supervised federated learning for medical image segmentation," *IEEE Transactions on Medical Imaging*, 2022.

[81] D. Bala, M. S. Hossain, M. A. Hossain, M. I. Abdullah, M. M. Rahman, B. Manavalan, N. Gu, M. S. Islam, and Z. Huang, "Monkeynet: A robust deep convolutional neural network for monkeypox disease detection and classification," *Neural Networks*, vol. 161, pp. 757–775, 2023.

[82] M. M. Ahsan, M. R. Uddin, M. Farjana, A. N. Sakib, K. A. Momin, and S. A. Luna, "Image data collection and implementation of deep learning-based model in detecting monkeypox disease using modified vgg16," *arXiv preprint arXiv:2206.01862*, 2022.

[83] A. S. Jaradat, R. E. Al Mamlook, N. Almakayeel, N. Alharbe, A. S. Almuflih, A. Nasayreh, H. Gharaibeh, M. Gharaibeh, A. Gharaibeh, and H. Bzizi, "Automated monkeypox skin lesion detection using deep learning and transfer learning techniques," *International Journal of Environmental Research and Public Health*, vol. 20, no. 5, p. 4422, 2023.

[84] V. H. Sahin, I. Oztel, and G. Yolcu Oztel, "Human monkeypox classification from skin lesion images with deep pre-trained network using mobile application," *Journal of Medical Systems*, vol. 46, no. 11, p. 79, 2022.

[85] Y. Yang, Q. J. Wu, X. Feng, and T. Akilan, "Recomputation of the dense layers for performance improvement of dcnn," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 11, pp. 2912–2925, 2019.

[86] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607.

[87] D. Kristine and P. Veiko, "Poor roads cost canadians $3 billion annually: Caa study," *Canadian Automobile Association*, 2021.

[88] S. Minaee, Y. Boykov, F. Porikli, A. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3523–3542, 2022.

[89] T. Akilan, Q. J. Wu, and W. Zhang, "Video foreground extraction using multi-view receptive field and encoder–decoder dcnn for traffic and surveillance applications," *IEEE Trans. Vehicular Technology*, vol. 68, no. 10, pp. 9478–9493, 2019.

[90] T. Zhang, D. Wang, and Y. Lu, "Ecsnet: An accelerated real-time image segmentation cnn architecture for pavement crack detection," *IEEE Trans. on Intelli. Transporta. Sys.*, vol. 24, no. 12, pp. 15 105–15 112, 2023.

[91] Y. Han and J. C. Ye, "Framing u-net via deep convolutional framelets: Application to sparse-view ct," *IEEE Trans. on Medical Imaging*, vol. 37, no. 6, pp. 1418–1429, 2018.

[92] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[93] C. Liu, C. Zhu, X. Xia, J. Zhao, and H. Long, "Ffedn: Feature fusion encoder decoder network for crack detection," *IEEE Trans. on Intelli. Transporta. Sys.*, vol. 23, no. 9, pp. 15 546–15 557, 2022.

[94] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," in *Intl. Conf. on Learning Representations*, 2018.

[95] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[96] X. Sun, Y. Xie, L. Jiang, Y. Cao, and B. Liu, "Dma-net: Deeplab with multi-scale attention for pavement crack segmentation," *IEEE Trans. on Intelli. Transporta. Sys.*, vol. 23, no. 10, pp. 18 392–18 403, 2022.

[97] J.-M. Guo, H. Markoni, and J.-D. Lee, "Barnet: Boundary aware refinement network for crack detection," *IEEE Trans. on Intelli. Transporta. Sys.*, vol. 23, no. 7, pp. 7343–7358, 2022.

[98] H. Wang, C. Liu, Y. Cai, L. Chen, and Y. Li, "Yolov8-qsd: An improved small object detection algorithm for autonomous vehicles based on yolov8," *IEEE Trans. on Instrumentation and Measurement*, vol. 73, pp. 1–16, 2024.

[99] C. Chen, C. Wang, B. Liu, C. He, L. Cong, and S. Wan, "Edge intelligence empowered vehicle detection and image segmentation for autonomous vehicles," *IEEE Trans. on Intelligent Transportation Systems*, vol. 24, no. 11, pp. 13 023–13 034, 2023.

[100] Y. Ma, J. Liu, Y. Liu, H. Fu, Y. Hu, J. Cheng, H. Qi, Y. Wu, J. Zhang, and Y. Zhao, "Structure and illumination constrained gan for medical image enhancement," *IEEE Trans. on Medical Imaging*, vol. 40, no. 12, pp. 3955–3967, 2021.

[101] K. Liu, Z. Ye, H. Guo, D. Cao, L. Chen, and F.-Y. Wang, "Fiss gan: A generative adversarial network for foggy image semantic segmentation," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 8, pp. 1428–1439, 2021.

[102] A. Abdollahi, B. Pradhan, G. Sharma, K. N. A. Maulud, and A. Alamri, "Improving road semantic segmentation using generative adversarial network," *IEEE Access*, vol. 9, pp. 64 381–64 392, 2021.

[103] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conf. on computer vision and pattern recognition*, 2016, pp. 3213–3223.

[104] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla, "Segmentation and recognition using structure from motion point clouds," in *Computer Vision–ECCV 2008: 10th European Conf. on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I 10*. Springer, 2008, pp. 44–57.

Appendix

Appendix A: IEEE Permission to Reprint

In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of Lakehead University's products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html and https://www.ieee.org/publications/rights/author-rights-responsibilities.html to learn how to obtain a License from RightsLink.

Appendix B: Source Code

The source codes of this thesis are available on GitHub.

For more information about the author's publications, please refer to Google Scholar and LinkedIn profiles.

Google Scholar: Google Scholar.

LinkedIn: LinkedIn.