A Semi-supervised Federated Learning Approach Leveraging Pseudo-labeling For Knee Osteoarthritis Severity Detection

by

Rakib Hossain Rifat
22366030

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
M.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
June 2024

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Rakib Hossain Rifat

22366030

Approval

The thesis titled "A Semi-supervised Federated Learning Approach Leveraging Pseudo-labeling For Knee Osteoarthritis Severity Detection" submitted by

1. Rakib Hossain Rifat (22366030)

Of Summer, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of M.Sc. in Computer Science in June 11, 2024.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Golam Robiul Alam
Professor
Department of Computer Science and Engineering
BRAC University

Examiner:
(External)

Dr. Shamim H Ripon
Professor
Department of Computer Science and Engineering
East West University

Examiner:
(Internal)

Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Sadek Ferdous
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Associate Professor; Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Within medical image analysis, appropriately classifying the extent of knee osteoarthritis is a significant obstacle, made more difficult by the scarcity of annotated data and strict privacy rules. Conventional approaches are hindered by the exorbitant expenses, limited availability of annotated datasets, as well as issues over the confidentiality of patient data. To overcome these challenges, we propose a method which is a Federated Learning Framework that utilizes pseudo-labeling we are calling it PLFL. Our innovative approach avoids the cost of human annotation and guarantees patient confidentiality through Federated Learning while reducing the dangers linked to adversarial assaults and annotation mistakes. Our proposed method works under the assumption that the server is the only custodian of gold label data, while the client side does not have any label data. The server utilizes gold-labeled data to train the global model and subsequently applies the federated learning approach. Clients add labels to unlabeled data by picking labels that meet or exceed a minimal threshold level of confidence in the prediction. Once data on the client side reaches the specified confidence score, it is added to the client's dataset. Upon receiving the labeled data, the client initiates the training process and sends the weight of the local model. Subsequently, the server aggregates the weights of each model using the FedAvg technique. The thorough assessment of our system, in comparison to the standard client-server-based Federated Learning approach (CSFL) and FixMatch-based semi-supervised Federated Learning (FSSFL) approach, clearly shows significant performance improvements. Our framework PLFL showed superior performance compared to other explained techniques, with consistent accuracy, weighted average precision, recall, and an F1-score of 0.88. Significantly, it outperforms both CSFL and FSSFL Frameworks, significantly enhancing model performance and efficiency. The proposed framework achieves an accuracy of 93.07% for the healthy class, 64.00% for the moderate class, and 100% for the severe class. Furthermore, our system has exceptional prediction precision, especially in detecting moderate and severe instances of osteoarthritis, surpassing rival frameworks. This is seen in the notable progress in accurately forecasting moderate and severe categories, highlighting the effectiveness of our method. The pseudo-labeling-based framework had the shortest duration for label generation and model training, 3.2 times shorter than the best-performing model of the traditional Federated Learning Framework (CSFL) and 1.7 times lower than the best-performing model of the FixMatch-Based Federated Learning Framework (FSSFL). This thesis proposes an innovative investigation into identifying knee osteoarthritis severity, the first instance of applying semi-supervised and federated learning approaches in this field. Our goal is to stimulate progress in medical image analysis by using our innovative technique, resulting in more precise diagnoses and better patient outcomes.

Keywords: Knee Osteoarthritis; Medical Image Analysis; Federated Learning; Pseudo-Labeling; Semi-Supervised Learning; Annotation Scarcity; Patient Confidentiality; Adversarial Attacks; Annotation Errors; Client-Server Architecture; Fix-Match Framework; Diagnosis Healthcare Data Privacy; AI in Healthcare; Computational Efficiency

Acknowledgement

First and foremost, I am grateful to Almighty Allah for the good health and well-being necessary to complete this thesis work. I extend my heartfelt thanks to my thesis supervisor, **Dr. Md. Golam Robiul Alam**. Without his encouragement, this thesis would never have been accomplished. His constant support, understanding, and constructive critiques over the year have greatly enriched my work.
I also take this opportunity to express my gratitude to all the respected faculty members of the department for their help and support. I thank my parents for their unceasing encouragement, support, and attention. Additionally, I am deeply thankful to my family for their unwavering support and patience throughout this journey. Finally, I am grateful to everyone who helped me in every possible way to make my thesis fruitful.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Over the last few decades, artificial intelligence more notably, machine learning has become a game-changing instrument in the medical field, transforming numerous areas involving administration, research, and practice. Machine learning algorithms have shown amazing potential to improve diagnosis, treatment planning, patient outcomes, and operational efficiency through the analysis of tremendous amounts of healthcare data. Analyzing medical imaging is one important field of progress. Radiographs, being the most often conducted radiological examination, hold significant importance as a modality that has been extensively studied for many purposes. Deep learning is the rapidly advancing domain of artificial intelligence that has been widely applied in several sectors, including the realm of medicine. Deep learning models in particular have demonstrated an unequalled ability to identify anomalies and diseases from medical imaging including CT scans, MRIs, and X-rays. Because algorithms using deep learning can precisely recognize patterns and traits that the human eye would miss, diseases including cancer, cardiovascular ailments, and neurological problems can be found early. Beyond diagnosis and therapy, machine learning has improved resource management and healthcare operations.

1.1 Motivation

Osteoarthritis (OA) can be seen as a worldwide problem. Osteoarthritis most commonly affects weight-bearing joints such as the knee, hip, and spine. Knee osteoarthritis(KOA), a common chronic joint illness, can be defined by gradual cartilage loss in the joint[58].In the United States, it is the most common joint condition, with symptomatic knee osteoarthritis affecting 10% of men and 13% of women aged 60 and older[6]. In India, Osteoarthritis is the second most prevalent rheumatologic issue and the most common joint condition, affecting between 22% to 39% of the population [3]. Approximately 45% of women aged 65 and older experience symptoms, whereas radiological evidence is detected in 70% of women in the same age group [1], [2]. Knee osteoarthritis can be caused by various factors such as aging, genetics, obesity, joint injuries, and excessive mechanical stress. Cartilage creates a low friction-bearing surface and allows for smooth joint movement. Reduction of this cartilage in the knee joint is the main reason behind KOA. This is a serious condition that affects people's every day activities by hampering their ability to walk. KOA is primarily responsible for the inability to move; if discovered in its later stages, full recovery is practically difficult, and knee replacement is the only alternative available, which is expensive. Early detection and treatment make it possible to maintain physical function, reduce pain, and slow disease progression, allowing people to live active lives. In general, radiographic imaging, such as X-rays, in addition to the patient's history and the results of the physical examination are used to diagnose knee osteoarthritis[41]. Radiography, specifically X-ray imaging, is a non-invasive, cost-effective, and readily accessible method that is highly valued for its ability to detect early signs and conduct mass screenings. However, its level of accuracy is lower. Magnetic Resonance Imaging (MRI) and dual-energy x-ray absorptiometry (DXA) are widely regarded as the most effective methods for detecting the severity of osteoarthritis (OA). However, they are expensive. With the recent major developments in X-ray imaging in medical fields, a wide range of anomalies and deformities in the body's muscles, limbs, and bones can now be detected. Using a variety of diagnostic methods, including radiography, MRI, gait analysis, and bioelectric impedance signals and radiographs are the most often used by expert doctors to diagnose knee osteoarthritis [35]. Using the Kellgren-Lawrence (KL) grading system, which classifies knee osteoarthritis into five stages from 0 (healthy) to 4 (severe) [4], one may frequently determine how bad the disease is. These grades are determined by severity and are as follows: 0 (healthy), 1 (doubtful), 2 (minimal), 3 (moderate), and 4 (severe) [18]. Measurement of joint space breadth and severity grading is made possible in large part by radiographs[52]. However, correctly determining these grades from radiographic images which might not have enough image enhancement can be difficult by deep learning models and mostly depends on the knowledge of medical specialists. However, the affordability and widespread availability of X-rays make them a promising tool for the early detection of osteoarthritis. By using deep learning models, it is possible to automate the precision and speed of early diagnosis.

With an eye on increased accuracy, effective resource allocation, and lower healthcare costs, researchers have explored a range of machine learning and deep learning approaches to enhance knee osteoarthritis detection over time. By supporting X-ray analysis and feature extraction, the use of image processing techniques including thresholding, masking, edge detection, and contrast enhancement improves the process. The growing amount of papers and Proceedings published in recent years, as 1.1 shows, is proof that researchers interest in addressing this issue.

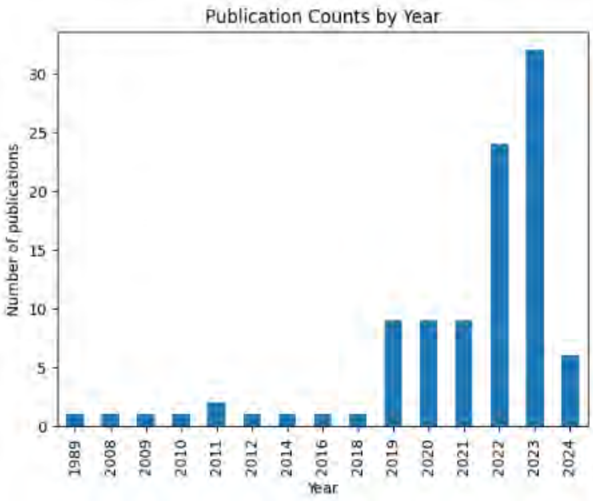


Figure 1.1: Number of Publication Count by Year on Knee Osteoarthritis.

Particularly in medical imaging, worries about data privacy endure despite the advancements in machine learning and deep learning. Medical data is delicate, containing details about medical disorders and past treatments, hence maintaining privacy and taking ethical issues into account is critical. Smooth access to medical data is hampered by the difficulty of preserving privacy.

1.2 Open challenges in Knee Osteoarthritis Diagnosis

A challenge for machine learning and deep learning practitioners is the availability of datasets. Consequently, there is a substantial body of research on the automated evaluation of osteoarthritis severity. However, the majority of the research relies on two publicly available datasets: the Osteoarthritis Initiative (OAI) [13] and the Multicentre Osteoarthritis Study (MOST) [8].In deep learning, providing more labeled data to a model typically yields better results. However, publicly available gold-label X-ray images are scarce. In the medical domain, finding reliable gold-label data is challenging due to several reasons. The complexity and variability of medical conditions and treatments make it difficult to establish a single "correct" label for many cases. Additionally, medical data are often sensitive and subject to privacy regulations, limiting access to large, well-labeled datasets. Labeling medical data is also time-consuming and expensive, requiring the expertise of trained professionals. These factors contribute to a scarcity of high-quality labeled datasets, which complicates the development of machine learning models in healthcare. Cai et al[48] highlighted a major problem: when end users label the data, there is a significant chance of making mistakes because they are not domain experts or lack interest. For that reason, researchers have attempted to address this issue by generating knee osteoarthritis X-ray images [44]. Nevertheless, these datasets are still not entirely reliable for the healthcare industry and individuals.

Another significant issue is that many hospitals possess such data, but privacy concerns and regulations prevent them from sharing it with researchers and the public. Which can be solved by using Federated learning. Federated learning is a decentralized machine learning approach where multiple devices collaboratively train a shared model while keeping data localized. It allows for model training without centralizing sensitive data, preserving privacy and security. This technique enables efficient learning across a network of devices, promoting scalability and adaptability in various applications[12]. Federated Learning (FL) holds significant promise in revolutionizing the medical domain by addressing critical challenges while safeguarding patient privacy and data security[49]. FL allows healthcare institutions to collaborate and train machine learning models collectively without the need to share sensitive patient data. This decentralized approach enables hospitals, research facilities, and pharmaceutical companies to pool their knowledge and resources while preserving the confidentiality of patient information.

Furthermore, a large portion of this data is still unlabeled because of a shortage of knowledge and qualified physicians. Data labeling is an involved process that calls for a great deal of experience. Federated learning and semi-supervised learning methods can be applied to get past these obstacles and make use of decentralized data. The medical industry presents several important problems for federated learning (FL) since labeled data is scarce. FL trains machine learning models cooperatively while maintaining local data from decentralized data sources like as hospitals or individual devices. However, there are frequently few big, varied, and well-labeled datasets in the medical profession spread over several locations or organizations. The complete potential of FL in healthcare is hampered by this dearth of labeled data. The lack of labeled data access makes FL less successful in training models or making good generalizations across many populations or situations. Furthermore impeding the sharing and aggregation of data required for FL are the sensitive character of medical data and privacy laws. Patient privacy and confidentiality are protected by HIPAA regulations, which require anonymization of data. Sensitive health information is protected and regulatory requirements are met when anonymization makes sure that people cannot be easily identified from the data[5]. In federated learning, allowing end users to label data opens up the possibility for adversaries to manipulate client data, potentially through actions such as altering labels or specific data features. These manipulations can result in data poisoning attacks, compromising the integrity and effectiveness of the learning process[34]. To address the shortage of labeled data in the medical domain, semi-supervised learning can be a viable approach that can work with a limited number of labeled data.

1.3 Our Proposed Solution: Overcoming the Problems in Knee Osteoarthritis Diagnosis and Medical Domain and its Effectiveness Comparison

In this study, we aim to address the challenges of insufficient supervised data, privacy concerns related to sharing medical data, and the issue of data poisoning. To tackle these challenges, we propose a semi-supervised learning approach that utilizes pseudo-labeling with zero-label data in federated clients, while relying on fully supervised data in the central server. Pseudo Contrastive Learning is a semi-supervised machine learning approach where models are designed to improve their performance by actively selecting and incorporating new, relevant data points for training. This iterative process allows the model to continuously adapt and improve its performance over time. In pseudo-labeling, labels are applied to unlabeled data by means of a model trained on both labeled and previously pseudo-labeled datasets. This process is repeated, in a self-training loop, adding newly tagged data to the training set[22] One machine learning paradigm called semi-supervised learning trains with a combination of labeled and unlabeled data. Particularly in cases when labeled data is scarce or costly to acquire, semi-supervised learning techniques can enhance the performance of the model by using the extra knowledge from the unlabeled data. Within Federated Learning (FL), semi-supervised learning can be rather important in improving the effectiveness and efficiency of the FL procedure. Through the active selection and incorporation of new, relevant data from decentralized sources, SSL can assist FL models in adapting to changing conditions and enhancing their performance over time by using the extra information from unlabeled data across decentralized sources. Particularly in cases when labeled data is limited or hard to come by from specific sources, this can assist the model to perform better.

Another semi-supervised method that can be utilized involves each client having a very small amount of labeled data. They can then label and other data and participate in training, known as FixMatch. FixMatch [26] is a pioneering approach in semi-supervised learning, innovatively merging the concepts of weak augmentation, strong augmentation, and pseudo-labeling to maximize the utilization of both labeled and unlabeled data. At its core, FixMatch employs weak augmentation, a gentle form of data transformation applied uniformly to both labeled and unlabeled data during training. This technique introduces subtle variations such as random cropping and horizontal flipping, facilitating better generalization without distorting the underlying content of the images. It also introduces the concept of strong augmentation, a more aggressive transformation strategy exclusively applied to the unlabeled data. These transformations, which include random rotations, translations, or changes in brightness, are designed to significantly alter the appearance of the images. By subjecting the model to such diverse and challenging inputs, strong augmentation encourages the learning of robust and invariant features. Through an iterative process, the model is trained using both the labeled data and the newly pseudo-labeled data, gradually improving its performance. FixMatch has been shown to achieve state-of-the-art results in scenarios where labeled data is scarce.

We will compare the results of traditional client-server-based Federated Learning (CSFL) and FixMatch-based Federated Learning (FSSFL) with the proposed method to demonstrate the efficacy of our approach. In addition, we will demonstrate that the combination of federated learning with semi-supervised approaches like as pseudo-labeling significantly improves the identification and categorization of knee osteoarthritis in its first phases using X-ray images. This work is the first known instance of using federated semi-supervised learning to classify knee osteoarthritis grade. In addition, our efforts have been directed toward mitigating data poisoning attacks and reducing labeling expenses through the use of semi-supervised learning. Furthermore, we have prioritized the need to safeguard patient data confidentiality through the utilization of federated learning methodologies.

1.4 Research Contributions

With a semi-supervised learning approach inside a federated learning framework, this thesis study attempts to address the difficulties of privacy and restricted data availability in medical imaging. Pseudo-labeling and zero-label data are combined in the solution for distributed clients. By use of pseudo-contrastive learning, the approach methodically selects and incorporates more data points into the training process. Furthermore lessening data poisoning assaults is the tactic. The economic efficiency of the method makes high-quality model training more accessible. The work shows potential in real-world healthcare applications by categorizing grades of knee osteoarthritis using X-ray images.

1. **An Improved Federated Learning Configuration** In the traditional federated learning configuration, the server does not engage in model training and does not possess any datasets, except for a small amount needed for model validation. The main duties of the server are selecting clients, distributing weights, and aggregating weights. In our updated federated learning arrangement, the data is still dispersed across clients, but the server possesses the gold label data. Firstly, the server trains a model using the provided gold label data and subsequently distributes the first model weights to the clients. After completing the initial training phase, the server takes on the conventional responsibilities of client selection, weight distribution, and weight consolidation in federated learning. By using this, we also ensure the following outcomes:

    - **Addressing concerns over privacy**
      This work ensures the privacy and security of sensitive patient information by utilizing a federated learning framework. This decentralized approach prevents the need to share medical data, thus safeguarding patient privacy while enabling collaborative learning across different institutions.

    - **Patient privacy through federated learning**
      Our research highlights the prioritization of federated learning approaches. By maintaining data decentralization and avoiding direct sharing, we protect patient privacy while yet facilitating the creation of robust machine-learning models.

2. **Advanced semi-supervised learning framework**
   We introduced an advanced semi-supervised learning framework that modifies the current pseudo-labeling approach, in the current pseudo-labeling approach [7] it is fully centralized and the model is trained by mixing pseudo-labels with existing annotated dataset and then retraining the model, but in our approach gold label data are trained only once in server and zero-label data in distributed clients. This modified methodology capitalizes on the advantages of both data privacy concerns and semi-supervised learning, hence improving the performance of the model even in situations when there is a limited amount of labeled data available. By utilizing this method, we additionally guarantee the following results:

    - **Resolving the issue of limited data availability**
      This work addresses the challenge of limited supervised data by employing

a semi-supervised learning technique. This approach reduces the requirement for abundant labeled data, making it feasible to train models with fewer annotations.

- **Data Poisoning Attack Mitigation**
  Our approach integrates tactics to reduce the risks linked to data poisoning attacks. By utilizing semi-supervised learning, we improve the resilience of the model against fraudulent data inputs, guaranteeing a higher level of reliability and security in the model's performance.

- **Cost-efficient labeling**
  The application of semi-supervised learning substantially decreases the costs associated with labeling by effectively leveraging both labeled and unlabeled data. This cost-efficient method enhances the accessibility of high-quality model training, particularly in fields where obtaining labeled data is costly.

3. **An In-depth Comparative Analysis**
   In this thesis, we not only introduce a new approach but also carry out a comprehensive comparative investigation. We evaluate our suggested method by comparing it to the client-server-based Federated Learning approach (CSFL) and the FixMatch-based semi-supervised Federated Learning (FSSFL). In order to guarantee the strength and significance of our comparisons, we employ various pre-trained models, such as DenseNet169, DenseNet201, and MobileNetV2. We conducted a comprehensive analysis of these frameworks, extensively assessing their performance, efficiency, and scalability. We specifically focused on identifying the strengths and drawbacks of each framework. This thorough assessment showcases the efficacy and adaptability of our methodology in various settings and model architectures.

1.5 Organization of the Report

The following text outlines the structure of this report: The Related Works are described in Chapter 2, and The background study for this project is outlined in Chapter 3. Chapter 4 provides a concise overview and examination of methodologies. Chapter 5 explores the discoveries and interpretation of the results. The main conclusion of the thesis is presented in Chapter 6.

Chapter 2

Related Work

2.1 Current Insights and Advancements in Knee Osteoarthritis

Researchers in the paper [50] focused on the fact that, due to the differences in body structures between individuals in different countries, using Western datasets would not be beneficial for Indians. Therefore, they utilized the Mendeley Dataset IV, which included Indian subjects and achieved an accuracy of 89% using EfficientNetB1. In another paper [43], using the OAI dataset, researchers focused on two things: first, they tried to detect the knee joint using YOLOv3, and then they used VGG16 for classification, and they achieved an accuracy of 89%. Researchers in a different study from 2023 [18] used an ensemble technique. They trained four different transfer learning methods ResNet-34, VGG-19, DenseNet 121, and DenseNet using the OAI dataset. Then, they put them all together to get better performance and got a 98% success rate. In this study [54], researchers tried to collect their own dataset using 2000 knee X-rays collected from a hospital, used CNN for feature extraction, and finally experimented with multiple machine learning models and got the highest accuracy of 90.1% using the K-Nearest Neighbour algorithm. Finally, in a 2022 study[42], the researchers tried a hybrid approach, a tri-weightage classification model that utilizes features from the x-ray image, questionnaire, and flexion angle, and finally, using ResNet-152v2 and Inception-ResNet-v2, they achieved an accuracy of 89.29%. A new approach called Siamese-GAP Network was introduced in a study [45] conducted in 2022. To be more precise, the Siamese network incorporates a series of Global Average Pooling (GAP) layers to extract information from each level. By employing YOLOv2[33], they conducted segmentation and performed binary classification to distinguish between the healthy and moderate classes, resulting in an accuracy of 88.38%.
A summary is given in the Table 2.1 below:

Table 2.1: Summary of the Literature Review on Knee Osteoarthritis

Research Work	Dataset	Model	Accuracy	Learning Philosophy	Learning Organization	Published Year
[50]	OAI dataset	EfficientNetB1	89%	Supervised Learning	Centralized	2023
[43]	OAI dataset	YOLOv3 for knee joint detection and VGG16 for classification	89%	Supervised Learning	Centralized	2022
[54]	OAI dataset	Ensemble method (ResNet-34, VGG-19, DenseNet 121, and DenseNet)	98%	Supervised Learning	Centralized	2023
[46]	Researchers collected	Feature extraction using CNN and ML methods	90.1%	Supervised Learning	Centralized	2022
[42]	OAI dataset	RestNet152V2 and InceptionResNetV2	89.29%	Supervised Learning	Centralized	2022
[45]	OAI dataset	Siamese-GAP Network	88.38%	Supervised Learning	Centralized	2022
[15]	Training set from the MOST validation set from the OAI	DNN	66.71%	Supervised Learning	Centralized	2018
[24]	Images from a hospital in Shanghai	Faster R-CNN	mAP 0.082	Supervised Learning	Centralized	2020
[17]	1024 knees images from the OAI dataset	Naive Bayes and RF classifiers	82.98%	Supervised Learning	Centralized	2019
[30]	18436 knees images from the MOST dataset	Textural ROI classification using CNN	AP 0.86	Supervised Learning	Centralized	2021
Our model	OAI Dataset	MobileNetV2	88.15%	Semi-supervised Learning	Federated and Cooperative	2024

From the table 2.1 and based on the analysis of various studies conducted between 2018 and 2023, it is evident that the majority of research on knee osteoarthritis has focused on the Osteoarthritis Initiative (OAI) [13] and the Multicentre Osteoarthritis Study (MOST) [8]. Specifically, there have been a total of seven studies conducted on the OAI and two studies conducted on the MOST. This indicates a shortage of annotated data available for knee osteoarthritis diagnosis. In two investigations, researchers attempted to acquire datasets from hospitals but were unable to publicly disseminate the data. The majority of researchers employed pre-trained image classification models such as EfficientNetB1 [28] and VGG16 [9]. Other studies employed a fusion of object detection and picture classification methods. Tariq et. al. [54] achieved the highest level of accuracy of 98% by employing a configuration that utilized four pre-training models in an ensemble-based approach.

Moreover, after conducting the current literature review, we discovered that most researchers used KL-Grading to check the severity of knee osteoarthritis, and the OAI dataset was the most commonly used. The common thing we can notice from the recently published papers is that most of the researchers have pointed out the unavailability of public data, therefore the gold label dataset is still a blocker for developing a good-performing model. Also, we need a generalized model with data from different regions, so there is still a lot of scope available for using semisupervised learning with federated learning.

Chapter 3

Background Studies

3.1 Federated Learning

Federated Learning is a distributed and privacy-preserving approach to machine learning. Consequently, the absence of a central database eliminates the necessity of storing all the sensitive data in one place, thereby preventing any potential data leaks. Federated Learning reverses the conventional approach of bringing the data to the machine learning model by instead bringing the machine learning model to the data. The training of the models is divided into sub-calculations that are carried out locally inside an organization. Upon completing the computations, only the anonymized (intermediate) outcomes are disclosed to the research organizations, excluding the privacy-sensitive data itself.
Federated Learning addresses two primary challenges in data analysis:

- Enhancing the quality of societal assessments,
- Protecting citizens' privacy rights.

Federated learning is a privacy-preserving method for training artificial intelligence models, ensuring that no one has access to or interacts with the data. It provides a means to leverage data for the development of new AI applications. A significant number of these AI applications were trained using data that was collected and processed in a single location. However, contemporary AI is transitioning towards a decentralized methodology. AI models are now being trained collectively on local devices such as mobile phones, laptops, or private servers, without the need for data to be transmitted elsewhere. Federated learning, a novel method of AI training, is increasingly being used as the industry standard to comply with a multitude of new requirements about the management and storage of sensitive data. Federated learning enables the utilization of raw data from many sources such as satellites, bridges, machines, and smart devices by processing it at its origin.

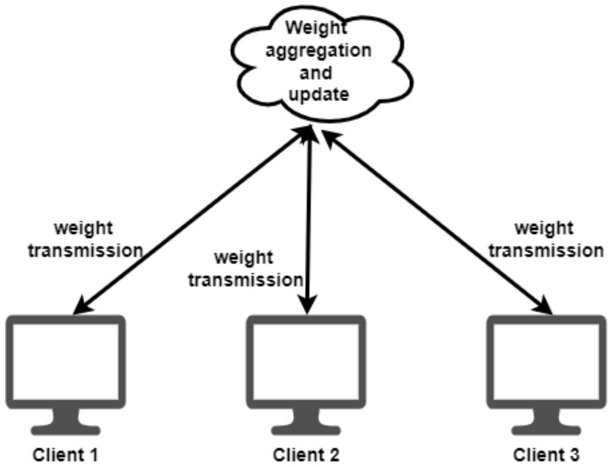


Figure 3.1: Working Procedure of Federated Learning

3.1.1 Current Insights in Federated Learning

In this study[19], the researchers pointed out the potential problem of data availability due to the data privacy regulations in the medical sector, which is a challenge for deep convolutional networks that require a large amount of data. They also state that high accuracy can be achieved by using federated learning and the proper weight aggregation method. They used a DNN model on the BraTS dataset to prove their claim. In a study of 2021 [40], Researchers tried to detect COVID-19 pneumonia, for this, they took the help of GAN for data generation, but due to data privacy and preventing reconstruction of data and keeping data privacy, they have proposed Federated Differentially Private Generative Adversarial Network (FedDP-GAN), using the (CXR)images dataset and FedDPGAN-based ResNe they achieved an accuracy of 94.45%. In another study [29] of COVID-19 case classification, researchers focused on data privacy and applied federated learning on (CXR)images dataset and compared it with traditional machine learning and found that federated learning outperforms the traditional machine learning method, where the traditional model with SGD optimizer got an accuracy of 94.82% fed-SGD achieved and accuracy of 95.96%. QAYYUM et al. [36] explored clustered federated learning for COVID-19 diagnosis using edge computing and also discussed challenges and technologies for deploying ML at the edge. Highlights collaborative learning framework for COVID-19 diagnosis leveraging clustered federated learning. And achieved a precision of 71% for detecting COVID-19, 97% for detecting healthy on the X-ray dataset, and a precision of 93% for COVID-19 and 86% for classifying healthy on Ultrasound images. Dou et al. [32] in their research Demonstrated federated learning for COVID-19 CT abnormalities detection. They utilized multinational datasets with internal datasets from three hospitals in Hong Kong and external validation on datasets from China and Germany. In their experiment, they found that larger training databases improve model performance on unseen datasets also collaboration across multiple clinical centers is crucial for AI system development. Furthermore, multicenter studies with collaborative efforts are valuable for handling data distribution. From our study, we have seen that Federated Learning (FL) approach is the emerging methodology for healthcare systems.

Many studies have used FL in medical studies such as MRI analysis[39], classifying prostate cancer [23], but mostly for COVID-19 diagnosis However, none of the mentioned studies have explored the possibility of Knee Osteoarthritis detection using Federated learning. Furthermore, it has been observed that the majority of research has concentrated on the key issue that, to enhance an AI model, data from various places is necessary. Hospitals have limitations in sharing data due to regulatory constraints and a shortage of skilled annotators. Consequently, there exists an opportunity for leveraging semi-supervised learning in conjunction with federated learning to solve this problem.

Table 3.1: Summary of the Literature Review on Federated Learning

Paper ID	Dataset	Method	Accuracy	Published Year	Used Semi-Supervised Learning
[19]	BraTS 2018	DNN model	Not mentioned	2022	No
[40]	(CXR) images dataset	FedDPGAN-based ResNet	94.45%	2021	No
[29]	(CXR)images dataset	Fed-SGD	95.96%	2021	No
[36]	X-ray and Ultrasound datasets	clustered federated learning (CFL)	The precision of 71% for COVID-19 and 97% for healthy on the X-ray dataset Precision of 93% for COVID-19 and 86% for healthy on Ultrasound dataset	2022	No

3.2 Semi-supervised learning

Semi-supervised learning occupies an intriguing position between supervised and unsupervised learning in the area of machine learning. In conventional machine learning, our typical practice involves utilizing a dataset that is completely labeled, with each example being assigned the right answer. But, in the real-world scenario, acquiring such a comprehensive dataset is frequently challenging, laborious, and costly. In contrast, unsupervised learning operates on unannotated data, aiming to identify patterns or clusters without any predefined criteria for these patterns. Semi-supervised learning bridges supervised learning and unsupervised learning techniques to solve their key challenges. With it, you train an initial model on a few labeled samples and then iteratively apply it to a greater number of unlabeled data.

3.2.1 Pseudo-labeling

Pseudo-labeling refers to the practice of augmenting the training data by using test data that has been anticipated with a high level of confidence. The pseudo-labeling strategy, initially introduced by [7] in 2013, involves utilizing a limited collection of labeled data in conjunction with a substantial amount of unlabeled data to enhance the performance of a model. The approach is really straightforward and consists of only four fundamental steps:

- Perform model training using a set of labeled data samples.

- Utilize the trained model to make predictions on a set of unmarked data.
- Utilize the anticipated labels to compute the loss on unlabeled data.
- Integrate the loss from labeled data with the loss from unlabeled data and perform backpropagation.

Pseudo-labels refer to assigning target classes to unlabeled data as if they were actual labels. The class with the highest predicted probability, determined by a network, is selected for each unlabeled sample (see Equation 3.1).

$$\text{Pseudo-labels} = \arg\max P(y = c|x) \tag{3.1}$$

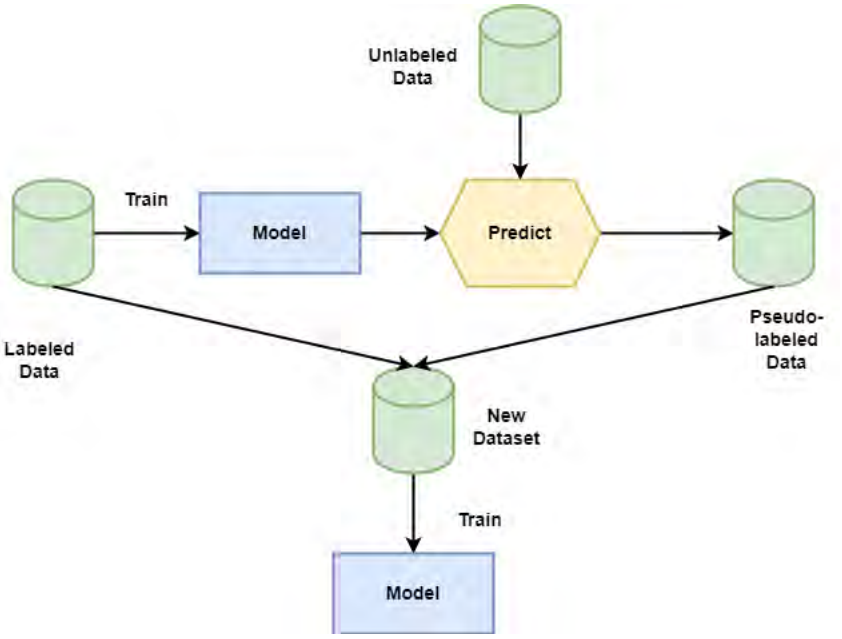


Figure 3.2: Working Procedure of Pseudo Labeling

Current Insights in Pseudo-labeling based Research Work

In this study[51], the researchers used Semi-supervised learning with pseudo-labeling for pancreatic cancer detection on CT scans they also addressed the challenge of detecting pancreatic cancer with limited labeled data. They have utilized a hybrid method combining pseudo-label and consistency regularization and found that Semi-supervised learning improves classification accuracy in pancreatic cancer detection. In another study[27], the researchers focus on learning from synthetic images for real-world applications. They also found that Active Pseudo-Labeling enhances model performance in semantic segmentation and detection and reduces domain gaps between synthetic and real images. Their method achieves an AP50 of 42.2 after fine-tuning which is an improvement in the object detection task on the benchmark dataset. In this research[53], the researchers addressed labeling challenges in e-health datasets due to high labeling costs or expertise requirements which is a blocker on the road to achieving high segmentation accuracy. So they proposed a Federated Semi-Supervised Learning model for medical image segmentation. They

Proposed a Federated Semi-Supervised Learning model for medical image segmentation. Using Federated Semi-Supervised Learning they Achieved the highest Dice scores of 89.23% and 91.95% in segmentation tasks which demonstrated significant improvements compared to state-of-the-art fundus image and prostate MRI segmentation. In another research [37], the researchers tried to detect plaques between two IVOCT datasets using Pseudo-label-based unsupervised domain adaptation techniques. They have found that label distribution learning improves the detection performance of unlabeled target images by correcting pseudo labels for vulnerable plaque detection. For VPS classification for IVOCT to HarbinOCT they achieved an F1 score of 89.45% and for HarbinOCT to IVOCT they achieved an F1 score of 85.02%

Most studies emphasize that label data is expensive to annotate and often unavailable owing to privacy constraints. No studies were discovered on the detection of knee osteoarthritis. However, one research did identify the difficulties associated with noisy label data caused by pseudo-labeling. An opportunity exists to utilize semi-supervised learning in combination with federated learning to address this issue.

3.2.2 FixMatch

FixMatch combines two SSL strategies: pseudo-labeling and consistency regularization. Its primary innovation stems from the combination of these two components and the consistency regularization process's usage of distinct weak and strong augmentations. The working procedure of FixMatch is shown in Figure 3.3.

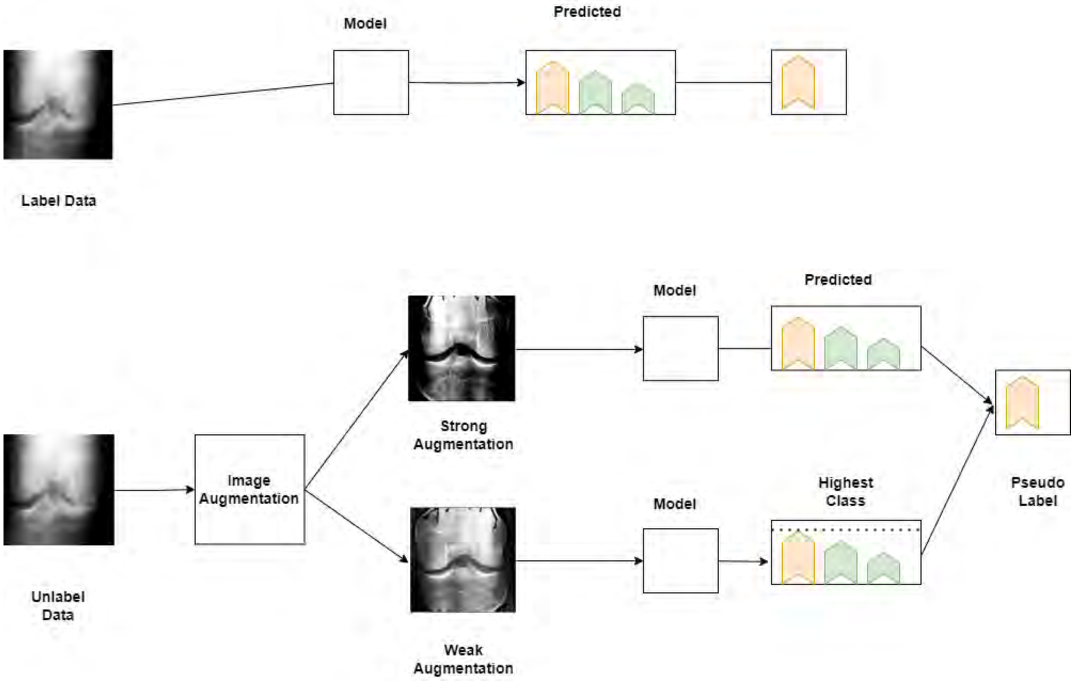


Figure 3.3: Working Procedure of FixMatch

Current Insights in FixMatch

Zhou et al. [56] were influenced by the noisy student method proposed in FixMatch-LS and a variant, FixMatch-LS-v2, where they were concerned about noisy training. They have also introduced the Kullback-Leibler loss. In these methods, they reduced the noisy pseudo-labels by introducing label smoothing to change the pseudo-label threshold. Using the ISIC 2018 and ISIC 2019 challenge datasets, they achieved an AUC of 91.63%, 93.70%, 94.46%, and 95.44% on the four proportions of labeled data from ISIC 2018. In another study, Ding et al. [31] tried to evaluate the quality of sinter, which is the main raw material of blast furnace ironmaking. FeO is an important indicator for evaluating the strength and reducibility of sinter. However, due to a lack of label data, they utilized FixMatch with Dense Net. Using the section image of the sintering machine tail, they have achieved an accuracy[57] introduced a novel concept of cross-pseudo-supervision by integrating self-training with consistency learning calling this model DFCPS which incorporates the concepts of Fixmatch. Using the Kvasir-SEG dataset they experimented with four settings using 1/2, 1/4, 1/8, and 1/16 of the labeled data and got mIoU of 80.12,77.42,76.53 and 72.39 respectively. DFCPS enhances the robustness and performance of the model. In another paper[25], researchers Investigate the MixMatch and FixMatch impact on histology images with noisy data in imbalanced settings. And found that MixMatch is more robust to imbalances compared to FixMatch. MixMatch has a higher average AUC in the imbalanced dysplasia class. One interesting finding of this study was Both methods degrade with a high level of imbalances. Yang [55] et al. took a closer look at the weak-to-strong consistency framework from FixMatch and then introduced the Unified Dual-Stream Perturbations approach (UniMatch) for superior results. They mainly focus on semi-supervised semantic segmentation using a weak-to-strong consistency framework and explore expanded perturbation space and dual-stream perturbation techniques for improvement. The results demonstrate superiority in remote sensing interpretation and medical image analysis. UniMatch also surpasses existing methods significantly across all evaluation protocols.

The medical domain, particularly disease classification, has not been thoroughly investigated using the semisupervised learning method FixMatch. However, researchers have observed the potential of FixMatch based on its algorithmic performance. Therefore, combining FixMatch with federated learning presents an opportunity to address the problem of insufficient labels. However, one research [47] has demonstrated that FixMatch does not yield significant improvements when tested on chest X-ray and retinal image datasets.

3.3 Federated Averaging

FedAvg [38] is one of the first and most often employed techniques for Federated Learning. During each cycle of training in FedAvg, a cohort of clients is chosen at random for the purpose of aggregation. During the process of aggregation, the parameters of each client are assigned weights and then averaged to create a global model. The weight assigned to each client is determined by the fraction of their data volume. It should be noted that in the implementation of FedAvg, more computation can be added to each client by doing numerous iterations of the local update before the averaging step. The equation for FedAvg can be represented as follows 3.2

The updated global model at time step $t+1$ is computed using the following equation:

$$\text{Global Model}_{t+1} = \sum_{i=1}^{C} \frac{N_i}{N} \times \text{Local Model}_i \tag{3.2}$$

where:

Global Model$_{t+1}$ is the updated global model at time step $t+1$.
C is the total number of clients.
N_i is the number of data points used for training by client i.
N is the total number of data points across all clients.
Local Model$_i$ is the model update from client i.

Explanation

- The term $\frac{N_i}{N}$ represents the proportion of data points contributed by client i relative to the total data points N.
- This proportion is used to weight the contribution of each client's local model update in the global model update.
- The sum $\sum_{i=1}^{C}$ aggregates these weighted contributions across all C clients.
- Dividing by C ensures that the global model update is averaged appropriately across all clients.

This formulation ensures that the global model update reflects the contributions of all clients proportionally to the amount of data they have used for training.

Chapter 4

Methodology

ur proposed method assumes that the server owns gold label data, while the client side does not have any label. The server trains the global model using gold-labeled data and then implements the federated learning technique. Clients assign labels to unlabeled data, selecting labels with minimum threshold level confidence or above in prediction. If an image achieves a confidence score, it is included in the client's data set. The client data set is divided into training, validation, and testing splits. After receiving newly labeled data, the client starts training and transmits the weight of the local model and the number of data points used for training to the server. The server then combines the weights of each model with the global model using the FedAvg method. If the result is unsatisfactory, the combined weights are transmitted to clients, who recommit the process of assigning labels and training local models. The training and communication illustrated in Figure 4.1

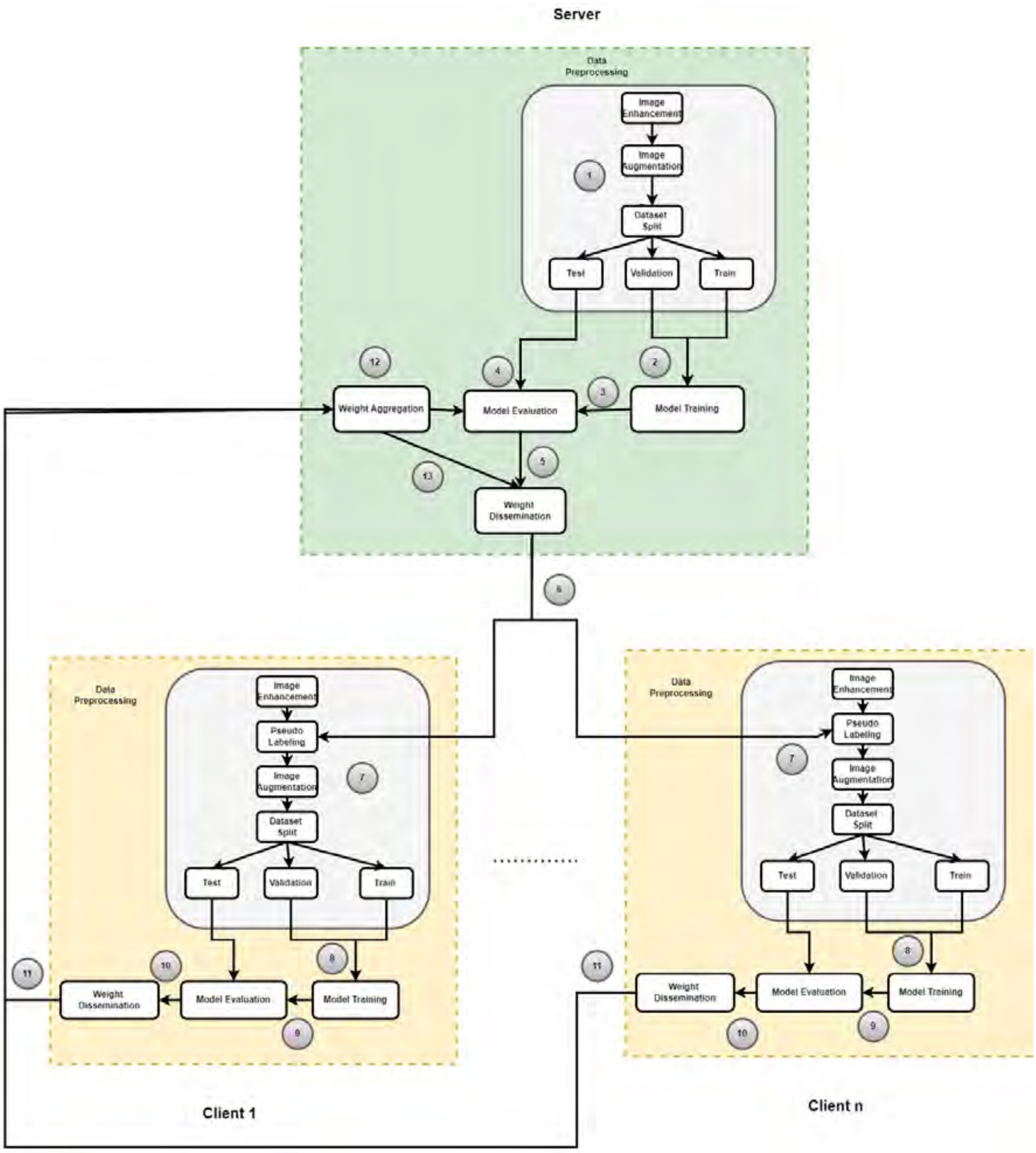


Figure 4.1: Top-Level Overview of the Proposed System

4.1 Dataset

In our whole experiment, we have used the Knee Osteoarthritis Severity Grading Dataset[13].

4.1.1 Knee Osteoarthritis Severity Grading Dataset

The University of Florida created the Knee Osteoarthritis Severity Grading Dataset. Osteoarthritis Initiative (OAI) organizes the photographs, which can be seen on Kaggle [13]. According to the Kellgren–Lawrence (KL) grading system, there are a total of 9786 knee images. These knee images are categorized into five severity levels: 0 (healthy), 1 (doubtful), 2 (minimal), 3 (moderate), and 4 (severe). The resolution of every single image was 224 pixels by 224 pixels. The healthy category comprised almost forty percent of the images in the dataset, while the doubtful category comprised approximately 18%, the minimal category comprised 26%, the moderate category comprised 13%, and the severe category comprised just over 3%. An overview of the dataset and some examples of its contents can be found in Table 4.1.

Table 4.1: Data Sample of Each Grade with Description

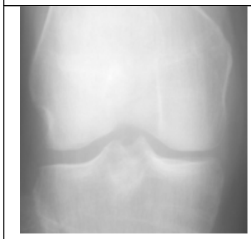
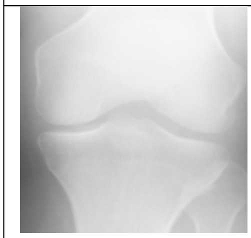
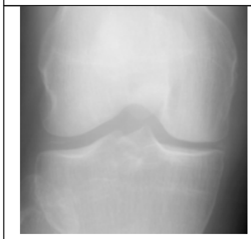
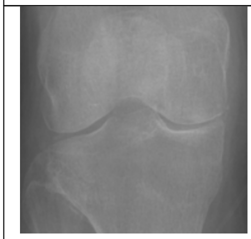
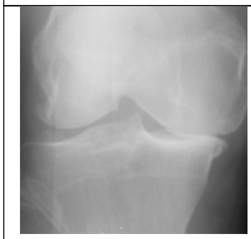
Data Sample	**Grade**	**Description**
	Grade 0 (Healthy)	An image of a knee indicating good health.
	Grade 1 (Doubtful)	Indications of potential joint narrowing with the presence of osteophytic lipping, though uncertainty exists.
	Grade 2 (Minimal)	Clearly identifiable osteophytes and potential narrowing of the joint space.
	Grade 3 (Moderate)	Presence of multiple osteophytes, definite joint space narrowing, and mild sclerosis.

Table 4.1: Data Sample of Each Grade with Description

Data Sample	Grade	Description
	Grade 4 (Severe)	Prominent osteophytes, significant joint narrowing, and severe sclerosis.

The data distribution for each class is illustrated in Figure 4.2.

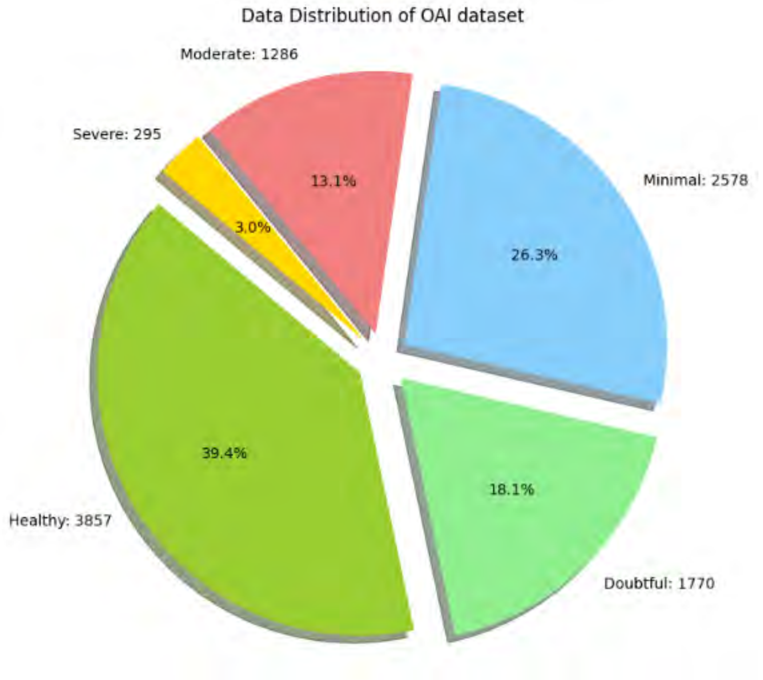
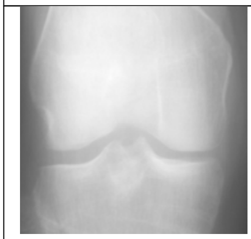
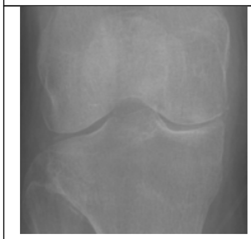
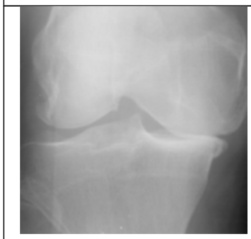


Figure 4.2: Data Distribution of Osteoarthritis Severity Grading Dataset

As we wanted to classify if a patient has knee Osteoarthritis or not so we have removed those classes that may contain knee Osteoarthritis like doubtful and minimal. To distinguish between the classes and degrees of severity and learn more distinguishable features, we have concluded that the Doubtful and Minimal classes should be removed from the list of classes. An overview of the final dataset, along with some examples of its content, can be found in Table 4.2

Table 4.2: Data Sample of Used Grades with Description

Data Sample	**Grade**	**Description**
	Grade 0 (Healthy)	An image of a knee indicating good health.
	Grade 3 (Moderate)	Presence of multiple osteophytes, definite joint space narrowing, and mild sclerosis.
	Grade 4 (Severe)	Prominent osteophytes, significant joint narrowing, and severe sclerosis.

The data distribution of each class follows after dropping the Doubtful and Minimal classes in Figure 4.3:

4.1.2 Data Preprocessing

To enhance the accuracy of the predictions, it is recommended that we make sure the photographs are of good quality and that models can easily learn from them. By doing this the model will be able to learn efficiently, recognize vital properties, and generate correct predictions as a result of this. Given this, we have decided to enhance the overall quality of the photographs. To improve the X-ray image and precisely identify the knee region, we made use of an image improvement approach that was implemented using OpenCV[60]. In the beginning, binary thresholding is applied in order to distinguish and separate the knee region. It is possible to

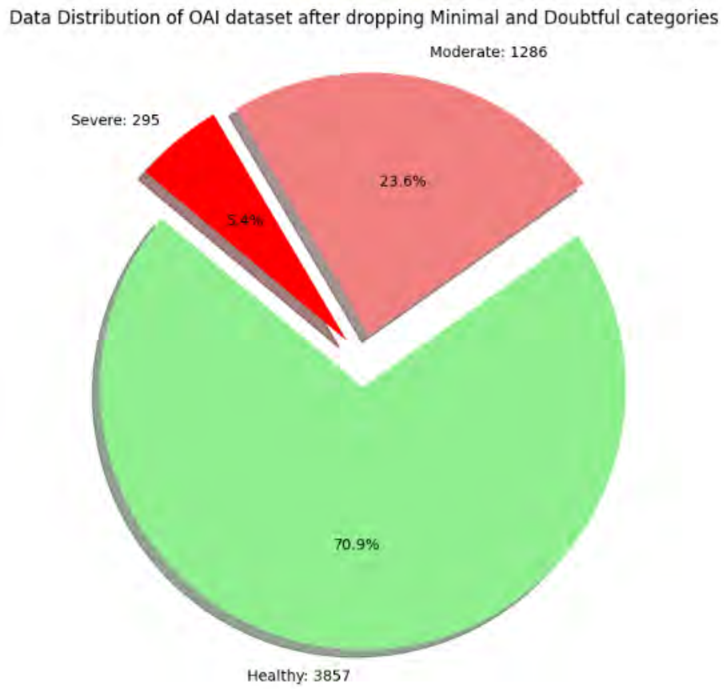


Figure 4.3: Data Distribution of Osteoarthritis Severity Grading Dataset Excluding Doubtful and Minimal Classes

construct a binary mask by identifying and making use of the principal contour that is specific to the knee. Following that, the grayscale picture is put through a bitwise AND operation using this mask in order to combine the two images. As a consequence of this, histogram equalization is utilized to enhance the contrast and visibility of the knee region. Table 4.3 includes explanations and methodology for carrying out the changes, while the graphic displays a transformation process that is carried out step by step.

Table 4.3: Step-by-Step Image Enhancement Procedure

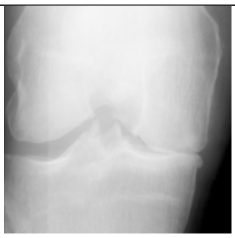
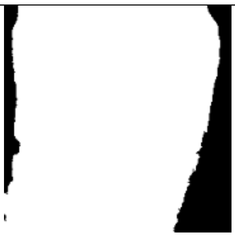
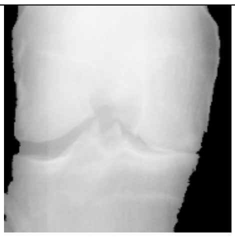
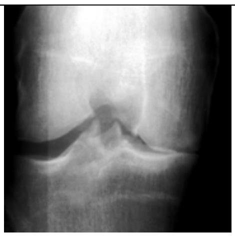
Step	Description	Sample Output
Convert to Grayscale	Convert the input color image to gray-scale.	
Gaussian Blur	Apply Gaussian Blur to the gray-scale image for noise reduction.	-
Adaptive Thresholding	Applies adaptive thresholding to the blurred image to create a binary image.	-

Table 4.3: Step-by-Step Image Enhancement Procedure

Step	Description	Sample Output
Find Contours	Finds contours in the binary image.	-
Get the Largest Contour	Find the largest contour (assumption: representing the knee).	-
Create Mask	Creates Mask of the largest contour	
Bit-wise AND Operation	Applies bit-wise AND operation between the gray-scale image and the mask to get the segmented knee area.	
Histogram Equalization	Applies histogram equalization to enhance the contrast of the segmented knee area.	

After data enhancement, we removed those images that were fully dark or half knee not visible, so we have gone through a manual checking and removed those images, here is a sample of the images we removed manually displayed in Figure 4.4, so after removing dark images, the data distribution became like this Figure 4.5:

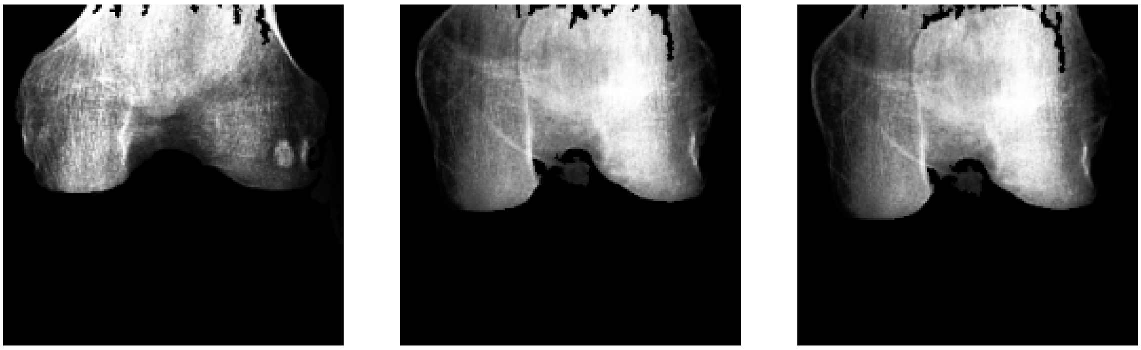


Figure 4.4: Sample Removed Images Where Part of the Knee is Not Visible

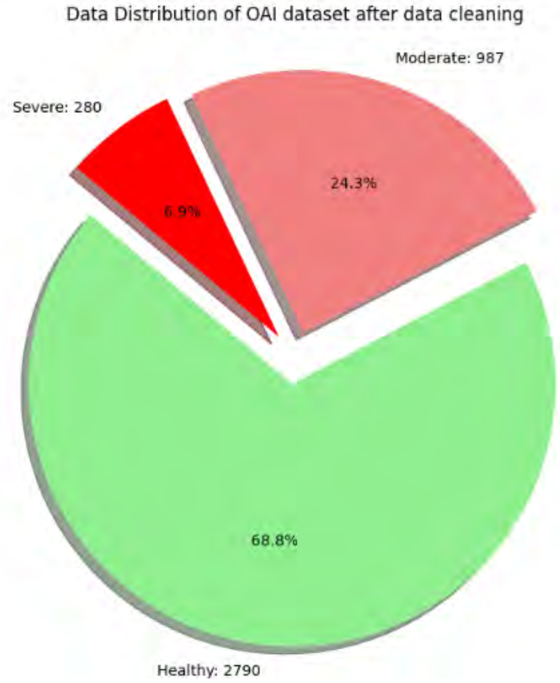


Figure 4.5: Data Distribution of Osteoarthritis Severity Grading Dataset After Removing Corrupt Images

4.1.3 Dataset Split

1. Dataset Split for Client-Server Based Federated Learning

Initially, we obtained the complete clean dataset consisting of 4057 photos. We then put aside 10% of this data only for testing reasons. This test data is specifically reserved for evaluating the performance of the aggregate model on the server. Subsequently, we divide the training dataset into two equal parts, assigning one half to client 1 and the other half to client 2.
The distribution of client and test data follows Table 4.4

Table 4.4: Class and Client-Wise Data Distribution for Federated Learning

Portion	Healthy	Moderate	Severe
Client 1	1255	444	126
Client 2	1264	439	123
Server	271	104	31

2. Dataset Split for Pseudo-labeled Federated Learning and FixMatch Federated Learning

At first, we took the complete dataset, which consisted of 4057 pictures. After that, we partitioned the complete dataset into three unique parts: one was put aside for client 1, another was designated for client 2, and the third and final component was assigned to the server.
The distribution of client and test data follows Table 4.5

Table 4.5: Class and Client-Wise Data Distribution for Pseudo-labeled Federated Learning and FixMatch Federated Learning

Portion	Healthy	Moderate	Severe
Client 1	930	329	93
Client 2	930	329	93
Server	930	329	93

4.1.4 Image Augmentation

Image Augmentation for Federated Learning and pseudo-labeled Federated Learning

Image data augmentation is creating additional variations of pictures in a given dataset by applying transformations, hence enhancing its variety. When it comes to implementing computer vision solutions that can be used effectively, it is better to have larger datasets that encompass all the visual characteristics of the item being targeted. However, implementing this is more challenging than simply expressing it verbally. Image data gathering necessitates the human acquisition and annotation of pictures, and it is unfeasible to record every conceivable circumstance that may be beneficial for the computer vision model. Data augmentation of image data reduces the amount of time required to create an ideal dataset by several person-hours. By safeguarding against overfitting, it enables us to enhance the performance of your model utilizing the available dataset.
We applied data augmentation techniques utilizing the Keras [59] module in TensorFlow [61] to enhance our dataset. The techniques involved in picture manipulation include resizing, rotating, adjusting the height and width, applying a shear, zooming, and horizontally flipping the images using a 'nearest' fill mode. The data augmentation method with the value and sample output is presented in Table 4.6

Table 4.6: Step-by-Step Image Augmentation Procedure

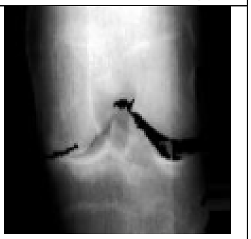
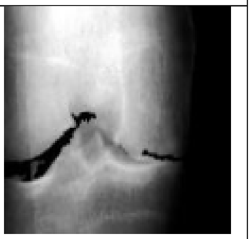
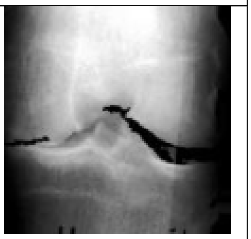
Parameter	Value	Description	Sample Image
Original Image	-	-	
Rescale	1.0 / 255	Rescale pixel values to [0, 1]	-
Rotation	20	Random rotation within ±20 degrees	
Width Shift	0.1	Random horizontal shift within 10% of image width	
Height Shift	0.1	Random vertical shift within 10% of image height	
Shear	0.2	Shear transformations	
Zoom	0.2	Random zoom within 20%	

Table 4.6: Step-by-Step Image Augmentation Procedure

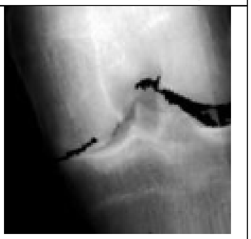
Parameter	Value	Description	Sample Image
Horizontal Flip	True	Randomly flip images horizontally	
Fill Mode	Nearest	Fill mode for handling newly created pixels (nearest neighbor)	-

Image Augmentation for FixMatch Federated Learning

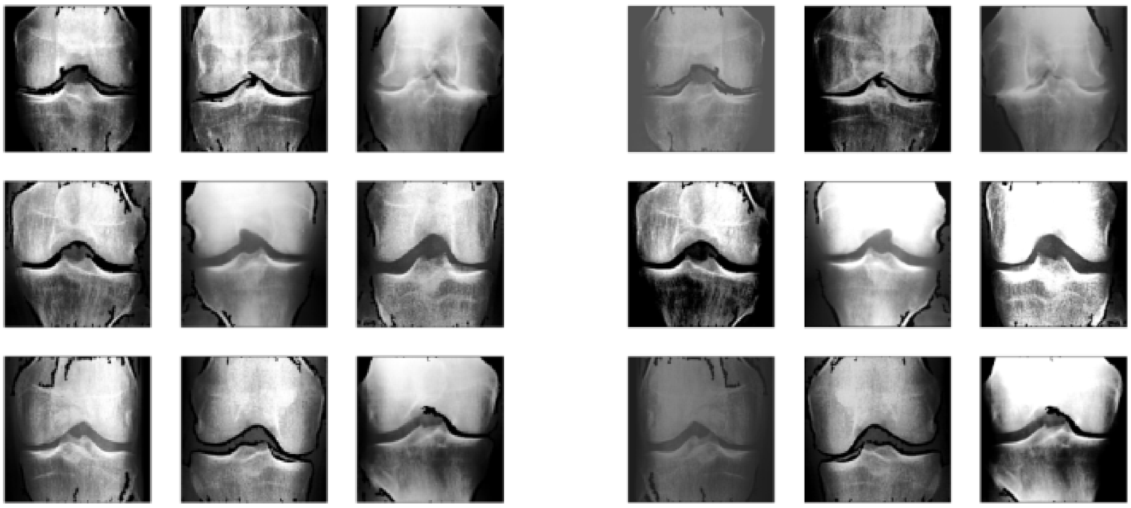
The idea behind FixMatch is derived from UDA [20] and ReMixMatch [16], and it makes use of two unique forms of augmentation: weak augmentation, which is used to generate pseudo-labels on unlabeled photos, and strong augmentation, which is used for prediction on unlabeled images. FixMatch uses both weak and powerful augmentations to achieve its desired level of efficacy.

1. **Weak Augmentation**
   A common variant of the flip-and-shift augmentation approach is known as weak augmentation. We applied these manipulations using random flips from left to right, random brightness, and random contrast. It is possible to find the values in Table 4.7

Table 4.7: Weak Augmentation Procedure with Description

Parameter	Value	Description
Random Flip Left to Right	-	Randomly flip images horizontally.
Random Brightness	max delta=0.2	Randomly adjust brightness by max delta 0.2.
Random Contrast	lower=0.2, upper=1.8	Randomly adjust contrast within range.

After applying weak augmentation our data became like Figure 4.6a.



(a) Original Images Before Weak Augmentation

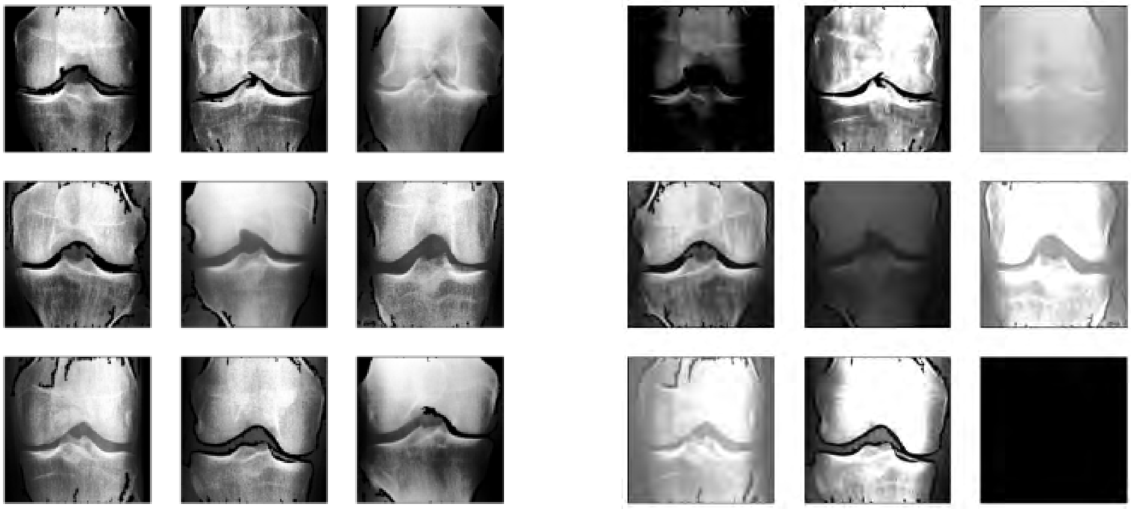
(b) Original Images After Weak Augmentation

Figure 4.6: Comparison of Original Images Before Augmentation and After Weak Augmentation

2. **Strong Augmentation**
   When these enhancements are applied, the resulting representations of the input photographs are very deformed. Several augmentations were applied to the image, including random flips, tweaks to brightness and contrast, changes to saturation, alterations to hue, and, as a last step, CutOut augmentation. A square section of the image is chosen randomly by the Cutout function, and then it is replaced with a color that is either a solid gray or a solid black. The values are contained in Table 4.8.

Table 4.8: Strong Augmentation Procedure with Description

Parameter	Value	Description
Random Flip Left to Right	Nil	Randomly flip images horizontally.
Random Brightness	max delta=0.8	Randomly adjust brightness by max delta 0.8.
Random Contrast	lower=0.2, upper=1.8	Randomly adjust contrast within range.
Random Saturation	lower=0.2, upper=1.8	Randomly adjust saturation within range.
Random Hue	max delta=0.2	Randomly adjust hue by max delta 0.2.



(a) Original Images Before Strong Augmentation

(b) Original Images After Strong Augmentation

Figure 4.7: Comparison of Original Images Before Augmentation and After Strong Augmentation

4.1.5 Models

1. DenseNet169

DenseNet[10] is a convolutional neural network architecture that incorporates dense connections between layers using dense blocks. In these blocks, all layers with the same feature-map sizes are directly connected. The DenseNet-169 model belongs to the collection of DenseNet models that are specifically developed for image classification tasks. The densenet169 model has a greater size, around 55MB. The DenseNet169 architecture consists of many sorts of layers, including convolutional, max pool, dense, and transition layers. In addition, the design employs two activation functions, specifically Relu and SoftMax. Every architecture is composed of four massive blocks, each with a different amount of layers. As an illustration, DenseNet-169 consists of layers arranged in the following sequence: [6, 12, 32, 32]. The convolution layer serves as the initial layer of the DenseNet model, which is responsible for the naming of Convolutional Neural Networks (CNN). DenseNet is a versatile framework that may be used for many computer vision tasks, such as image classification, object recognition, and semantic segmentation. We have utilized the DenseNet169 architecture from TensorFlow's Keras library [62] for the purpose of classifying the three severity classes in our study

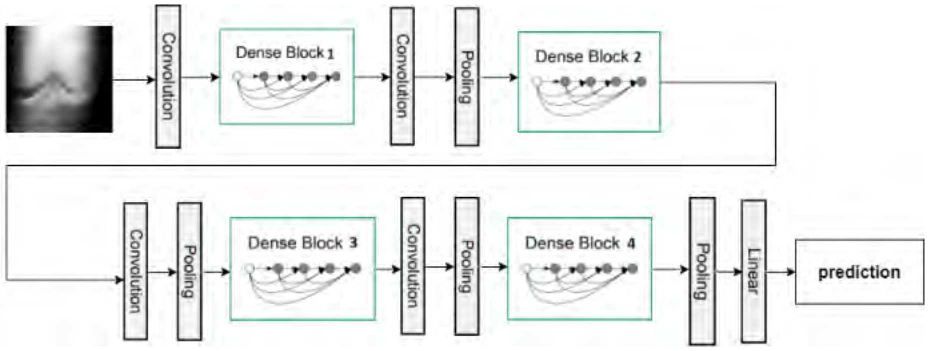


Figure 4.8: Architecture of DenseNet-169

2. MobileNetv2

Google is the designer of MobileNetV2[11], [14]. MobileNetV2 is a superior module that incorporates an inverted residual structure. The removal of non-linearities in thin layers is evident when compared with MobileNetV1[42]. By utilizing MobileNetV2 as the foundation for extracting features, exceptional results are also attained in object identification and semantic segmentation. MobileNetV2 consists of two distinct sorts of blocks. A residual block with a stride of 1 is present. Another option is to use a shrinking block with a stride of 2. Both sorts of blocks consist of three layers. The first layer consists of a 1×1 convolution operation using the ReLU6 activation function. The depthwise convolution corresponds to the second layer. The third layer consists of a 1×1 convolution operation, which does not include any non-linear activation function. It is asserted that when ReLU is applied repeatedly, deep networks only exhibit the capabilities of a linear classifier on the portion of the output domain that is not zero. Mobilenetv2 It is renowned for its efficiency and low computational demands and offers an ideal solution for medical image analysis, enabling robust diagnostic classification while minimizing resource usage. Using MobileNetV2 architecture from TensorFlow's Keras library [64] this architecture has been implemented.

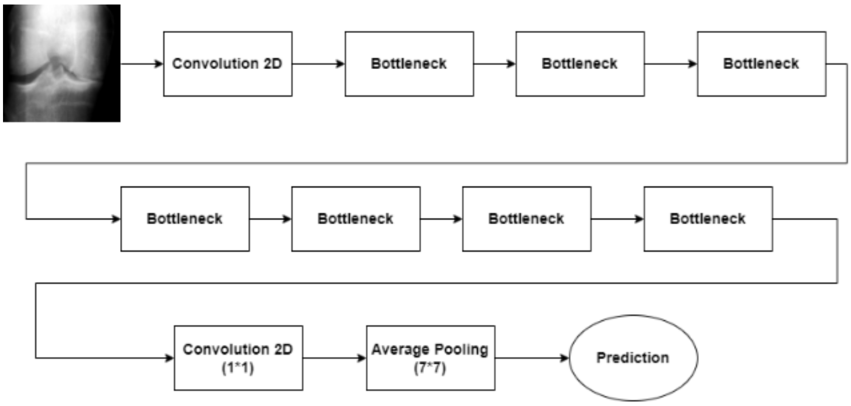


Figure 4.9: Architecture of MobileNetV2

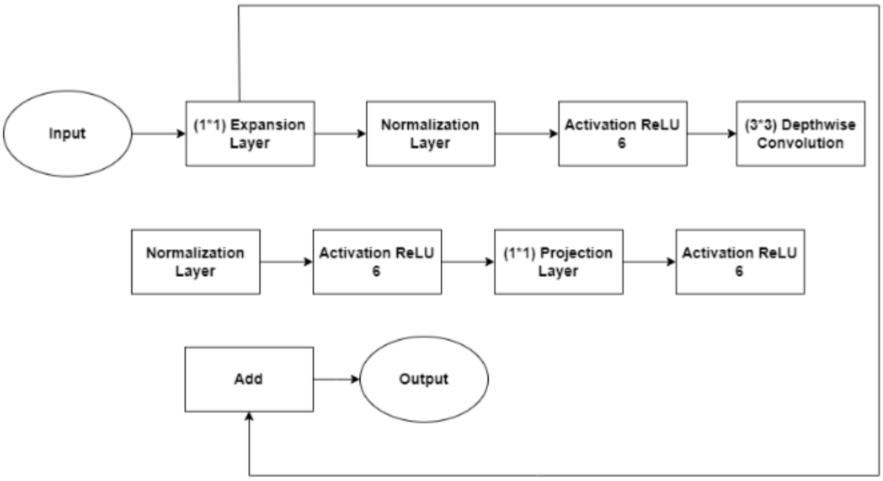


Figure 4.10: Architecture of Bottleneck of MobileNetV2

3. DenseNet201

The Dense Convolutional Network (DenseNet)[10] establishes connections between each layer and every other layer in a feed-forward manner. They mitigate the issue of the vanishing gradient, enhance the propagation of features, promote the reuse of features, and significantly decrease the parameter count.DenseNet operates on the principle that convolutional networks may achieve greater depth, accuracy, and training efficiency by including shorter connections between layers near the input and those near the output.DenseNet201 is characterized by its 201 layers, which results in a greater number of parameters and computational complexity compared to DenseNet169, which has 169 layers. DenseNet201 often has a greater number of parameters since it has a bigger depth and breadth in comparison to DenseNet169. This can result in possibly enhanced performance, particularly when dealing with bigger datasets, but it also necessitates a greater allocation of processing resources. The computational cost of DenseNet201 is higher compared to DenseNet169, mostly because of its bigger parameter count and deeper architectural complexity. We have utilized the DenseNet201 architecture from TensorFlow's Keras library [63] for the purpose of classifying the three severity classes in our study.

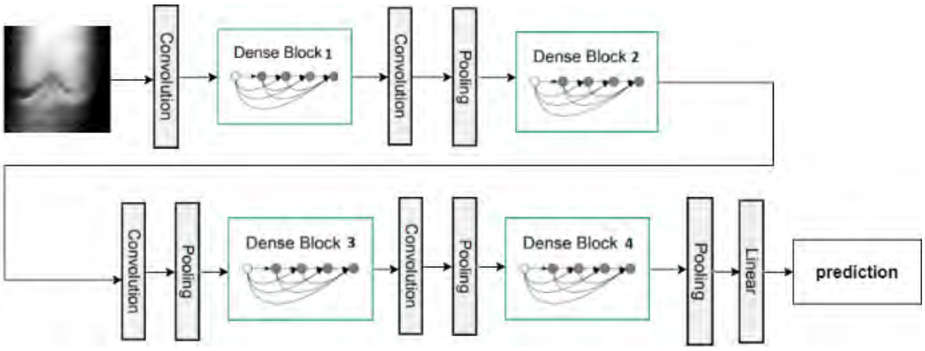


Figure 4.11: Architecture of DenseNet-201

4.1.6 Model Size and Parameters Comparison

The tables 4.9 and 4.10 present a detailed comparison of three neural network models—DenseNet169, DenseNet201, and MobileNetV2—focusing on their parameters and sizes.

DenseNet201 has the most parameters with 24,085,507 trainable and 18,321,984 non-trainable, totaling 42,407,491. DenseNet169 follows with 20,874,243 trainable and 12,642,880 non-trainable parameters, adding up to 33,517,123. MobileNetV2 is the smallest, having 16,057,347 trainable and 2,257,984 non-trainable parameters, for a total of 18,315,331.

Regarding size, DenseNet201 is the largest, with 91.88 MB for trainable parameters and 69.89 MB for non-trainable ones, totaling 161.77 MB. DenseNet169 has 79.63 MB of trainable and 48.23 MB of non-trainable parameters, amounting to 127.86 MB. MobileNetV2 is the most compact, with 61.25 MB for trainable and 8.61 MB for non-trainable parameters, totaling 69.87 MB.

DenseNet models, with their larger parameter sets and sizes, are likely to offer higher performance but require more computational resources. In contrast, MobileNetV2 is optimized for efficiency, making it suitable for environments with limited resources.

Table 4.9: Comparison of Models Based on Parameters

Model Name	Trainable Parameters	Non-trainable Parameters	Total Parameters
DenseNet169	20874243	12642880	33517123
DenseNet201	24085507	18321984	42407491
MobileNetV2	16057347	2257984	18315331

Table 4.10: Comparison of Models Based on Size

Model Name	Trainable Parameters Size (MB)	Non-trainable Parameters Size (MB)	Total Parameters Size (MB)
DenseNet169	79.63	48.23	127.86
DenseNet201	91.88	69.89	161.77
MobileNetV2	61.25	8.61	69.87

4.2 Proposed Method: Pseudo-labeling-Based Federated Learning Framework (PLFL)

Under this approach, we assume that the server only owns the label data. Based on our distribution, we have 930 images of healthy knees, 329 images of knees with moderate osteoarthritis, and 93 images of knees with severe osteoarthritis. These data are classified as our gold label data, whereas all the data on the client side has no label. Our approach started by training the server using the gold-labeled data. After training the global model for 20 epochs. We begin the implementation of the federated learning technique. Initially, we transmit this train model weights to the clients. Upon obtaining the weight of the global model, the clients started the task of assigning labels to the data that had not been previously labeled. During this stage, we selected only the labels that had a confidence level of 70% or above in the prediction. If an image achieves a confidence score of 70% on its outcome, we categorize it as belonging to that specific class and include this data to train the client model. Once the entire client dataset has been predicted, we proceed to divide it into separate halves for training, validation, and testing of the client model. After receiving newly labeled data, the client commences training. Once the training is complete, the client transmits the weight of the local model and the number of data points utilized for training to the server. Upon receiving the data from both clients, the server commences the process of combining the weights of each model with the global model that it had trained earlier using the FedAvg method. Next, the server evaluates the performance of this model using the separate test data that was set aside specifically for testing purposes. If the outcome is deemed unsatisfactory, the server proceeds to transmit the combined weights to the clients, who in turn recommence the process of assigning labels to their data and training local models. Subsequently, the clients submit the updated weights back to the server. Here we present the algorithmic overview of our method

Algorithm 1 Federated Learning Algorithm

```
Server Training:
Train a global model using the gold-labeled data.
Federated Learning Training:
for each iteration do
    Client Labeling:
    Receive the current global model weights.
    Clients label their unlabeled data based on the model predictions, considering only predictions with a confidence level of 70% or higher.
    Client Training:
    Clients split their labeled data into training, validation, and testing sets.
    Clients train their local models using the labeled training data.
    Client Model Transmission:
    Clients send their trained model weights and the number of data points used for training back to the server.
    Server Aggregation:
    The server aggregates the received model weights from all clients using Federated Averaging (FedAvg).
    Server Evaluation:
    The server evaluates the performance of the aggregated model using separate test data.
    Check Performance:
    if the model performance is deemed unsatisfactory then
        The server sends the aggregated weights back to the clients.
        Clients repeat the process of labeling their data and training local models.
    else
        The process stops.
    end if
end for
```

Client Distribution

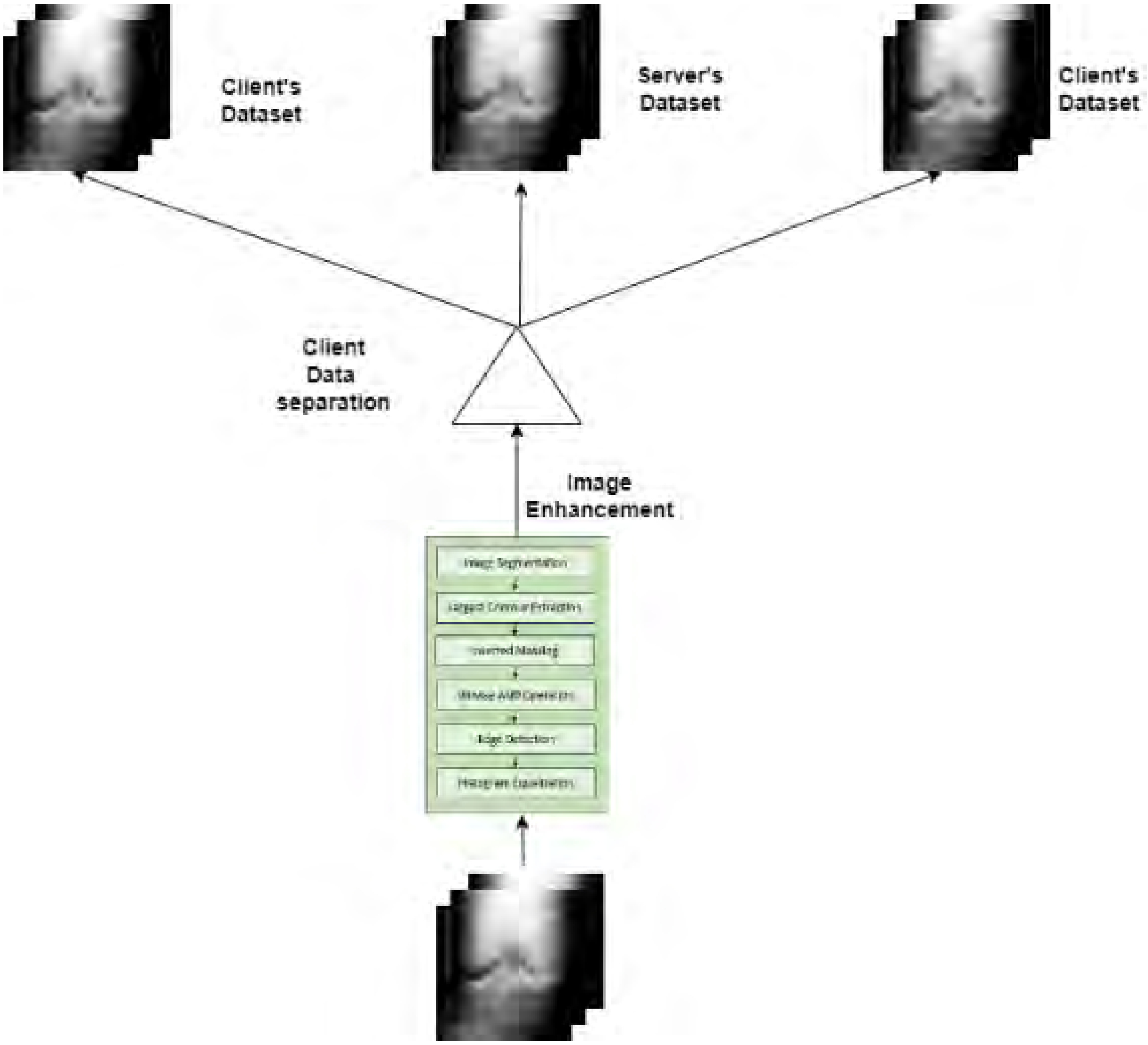


Figure 4.12: Client and Server's Dataset Creation of Pseudo-labeling-Based Federated Learning Framework

Method Overview

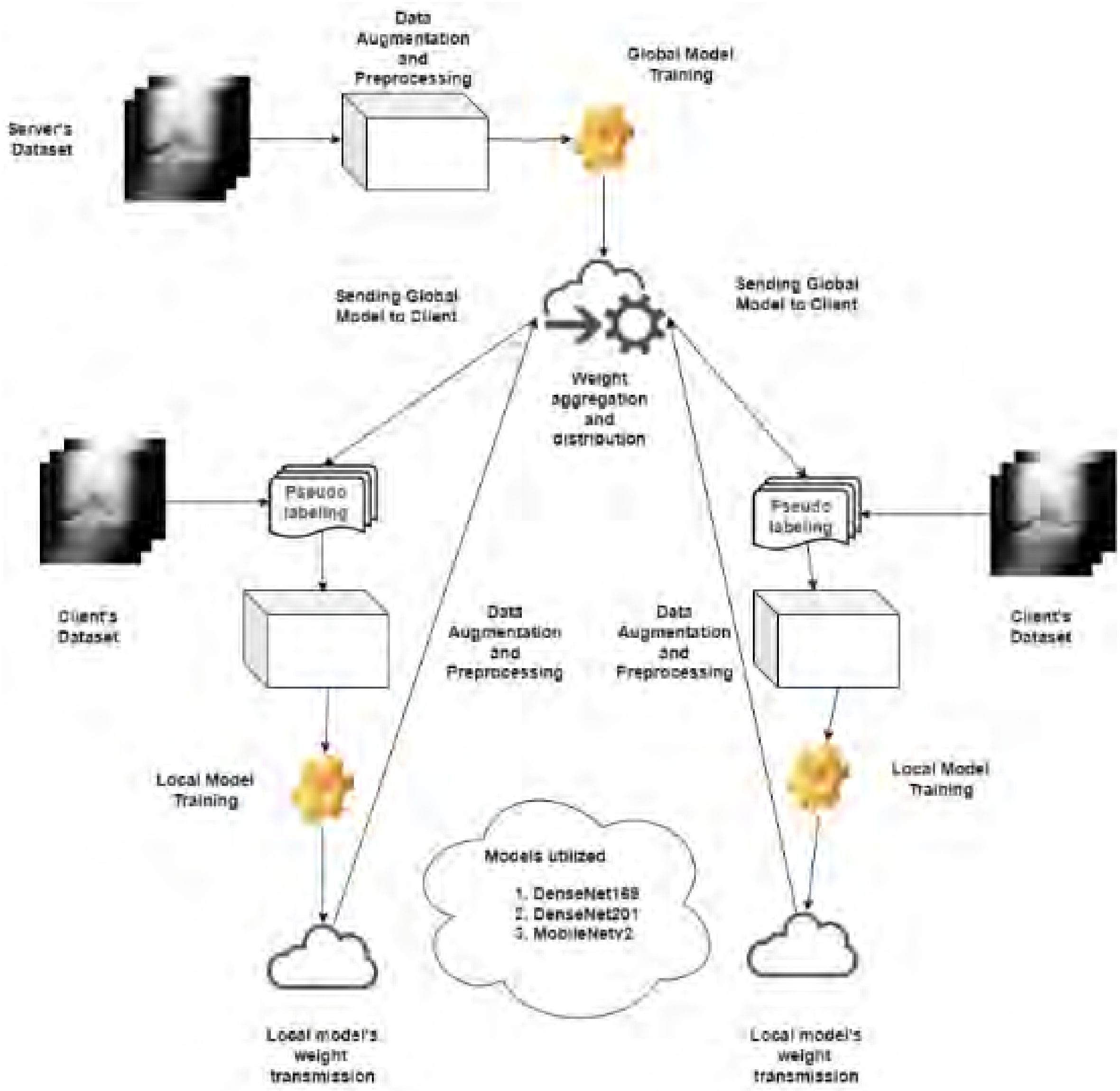

Data Augmentation and Preprocessing
Global Model Training
Server's Dataset
Sending Global Model to Client
Sending Global Model to Client
Weight aggregation and distribution
Pseudo labeling
Pseudo labeling
Client's Dataset
Client's Dataset
Data Augmentation and Preprocessing
Data Augmentation and Preprocessing
Local Model Training
Local Model Training
Models utilized
1. DenseNet169
2. DenseNet201
3. MobileNetv2
Local model's weight transmission
Local model's weight transmission

Figure 4.13: Pseudo-labeling-Based Federated Learning Framework.

4.3 Comparative Methods: Client-Server-Based Federated Learning (CSFL) and FixMatch-Based Federated Learning Framework (FSSFL)

4.3.1 Client-Server-Based Federated Learning Framework

The initial experiment is a Federated Learning framework that operates on a client-server architecture. This framework comprises two separate clients and a central server. The server has the responsibility of transmitting weights to the clients, receiving weights from the clients, and aggregating the weights. At first, the server distributes randomized weights to the clients. Upon getting the weights from the servers, the clients begin training their own local data. The client's data was divided into three segments: 80% for training the model, 10% for validation during the model training, and 10% for measuring the client's performance. Upon completing the data training process, each client transmits the weight and the number of samples utilized to the server. The server remains in a state of readiness to accept the weights and information from both clients. Upon receiving the weights from both clients, the server commences the process of aggregating the weights into a global model. The weighted FedAvg approach is utilized to aggregate the weights into a global model. Subsequently, the server evaluates the performance of the global model using data that has been set aside solely for testing. The server continuously executes this procedure until it attains a satisfactory outcome. Our experiment consisted of three communication rounds. In order to get high precision using the global model.
Here we present the algorithmic overview of our method:

Algorithm 2 Client-Server-Based Federated Learning Framework

```
Initialization:
Setup a server and multiple clients.
Initialize the global model on the server.
Distribute initial model weights to all clients.
Training Rounds:
while termination condition not met do
    Local Training:
    Clients receive the current global model weights.
    Each client trains its local model using its own data.
    Trained model weights are sent back to the server.
    Aggregation:
    Server collects and aggregates trained model weights from all clients.
    Aggregation is performed using weighted FedAvg.
    Evaluation:
    Server evaluates the performance of the global model using test data.
end while
Termination:
Repeat the training rounds for a predefined number of iterations or until convergence.
```

Overview of the Method

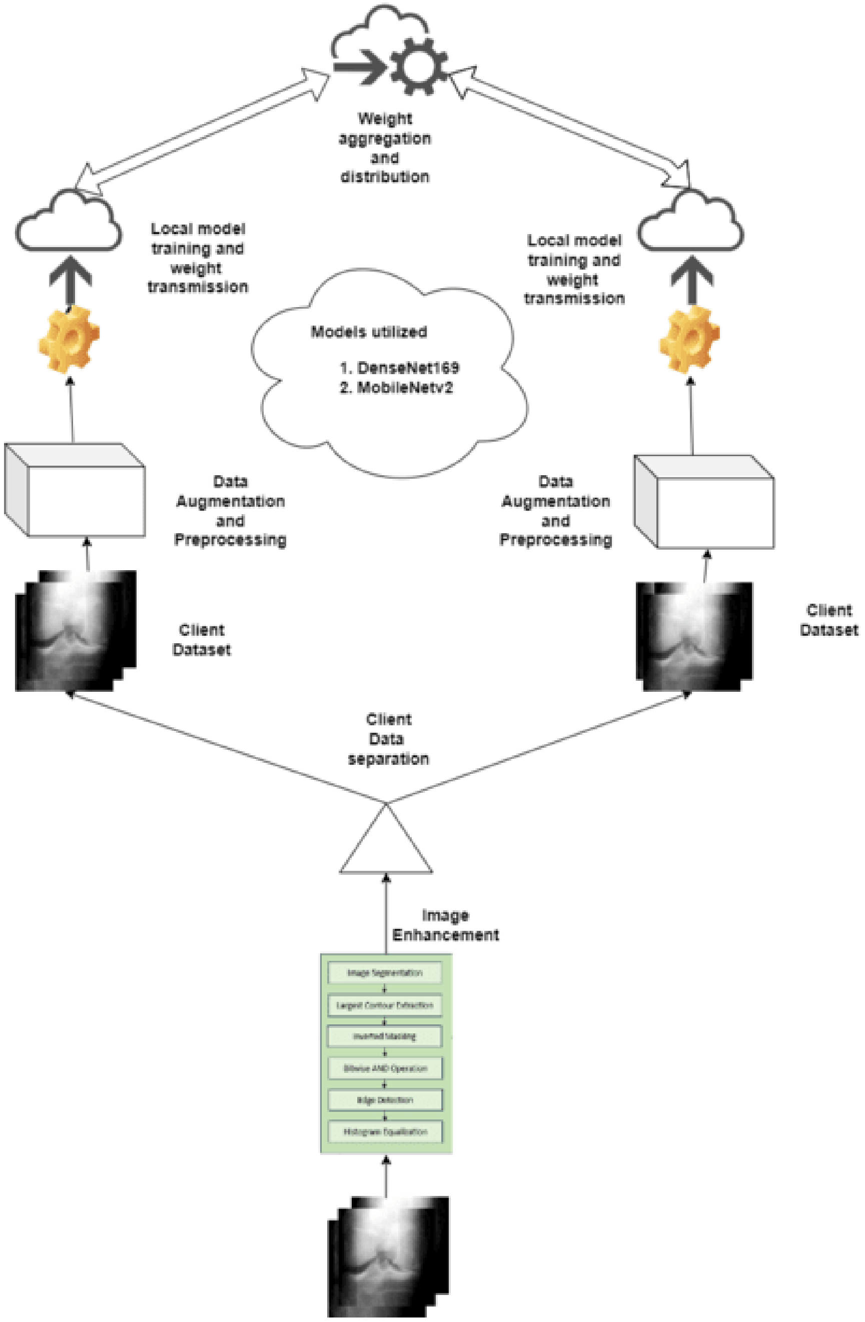

Weight aggregation and distribution
Local model training and weight transmission
Local model training and weight transmission
Models utilized
1. DenseNet169
2. MobileNetv2
Data Augmentation and Preprocessing
Data Augmentation and Preprocessing
Client Dataset
Client Dataset
Client Data separation
Image Enhancement
Image Segmentation
Largest Contour Detection
Inverted Masking
Bitwise AND Operation
Edge Detection
Histogram Equalization

Figure 4.14: Client-Server-Based Federated Learning Framework

4.3.2 FixMatch-Based Federated Learning Framework

This structure consists of two distinct clients and a central server. The server is responsible for providing weights to the clients, receiving weights from the clients, and aggregating the weights. Initially, the server assigns random weights to the clients. Upon the beginning of client training, we undertake the conversion of photos into Numpy format to enhance augmentation and processing capabilities. Initially, we allocated 20% of the data specifically for each clients, which would be exclusively utilized for testing the local model. We have determined that the remaining 80% of the data will be utilized for training the client. Within this 80%, 80% is unlabeled and 20% is labeled. In each batch, we include both labeled and unlabeled data, with the quantity of unlabeled data being twice as much as the labeled data. The model commences training by utilizing the labeled data and enhancing the unlabeled data through two methods: strong augmentation and weak augmentation. We utilize a supervised model to train our labeled pictures using cross-entropy loss. Two pictures are obtained for each unlabeled image by applying weak augmentation and strong augmentation. The image that has been enhanced with additional features is inputted into our model, and we obtain predictions for different classes. Here, the likelihood of the most confident class is being compared to a threshold of 70%. If the value exceeds the specified threshold, we consider that class as the ground label, also known as the pseudo-label. Next, the highly enhanced picture is inputted into our model to obtain a forecast across different classes. The cross-entropy loss is utilized to compare the probability distribution with a ground-truth pseudo-label. The losses are aggregated and the model is fine-tuned. After finishing the data training process, each client sends the weight and the number of samples used to the server. The server stays in a state of preparedness to receive the weights and information from both clients. After receiving the weights from both clients, the server starts the process of combining the weights into a global model. The weighted Federated Averaging (FedAvg) technique is employed to combine the weights and create a global model. Afterward, the server assesses the effectiveness of the global model by utilizing data that has been exclusively reserved for testing purposes. The server iteratively runs this operation until it achieves a desirable result. The experiment included two communication rounds. In order to get high precision using the global model.

Here we present the algorithmic overview of our method

Algorithm 3 Federated Learning with Data Augmentation

```
Initialization:
Assign random weights to the clients' models by the server.
Data Preparation:
Convert images into Numpy format for better augmentation and processing capabilities.
Allocate 20% of the data for testing each client's local model.
Reserve the remaining 80% for training, with 80% unlabeled and 20% labeled.
Training Process:
for each client do
    Train the client's model on the labeled data and augment the unlabeled data through strong and weak augmentation.
    Apply supervised learning on labeled images using cross-entropy loss.
    for each unlabeled image do
        Generate two augmented images (weak and strong augmentation).
        Predict the class probabilities for each augmented image.
        if the most confident class probability exceeds 70% then
            Assign it as the pseudo-label.
        end if
    end for
    Utilize cross-entropy loss to compare the probability distribution with the pseudo-label.
    Aggregate losses and fine-tune the model.
end for
Model Transmission:
After training, each client sends its model weights and the number of samples used to the server.
Server Aggregation:
The server receives weights from all clients and combines them into a global model using weighted Federated Averaging (FedAvg).
Model Evaluation:
Evaluate the effectiveness of the global model using reserved test data.
Iteration:
Iteratively repeat the process until a desirable result is achieved.
```

Client Distribution

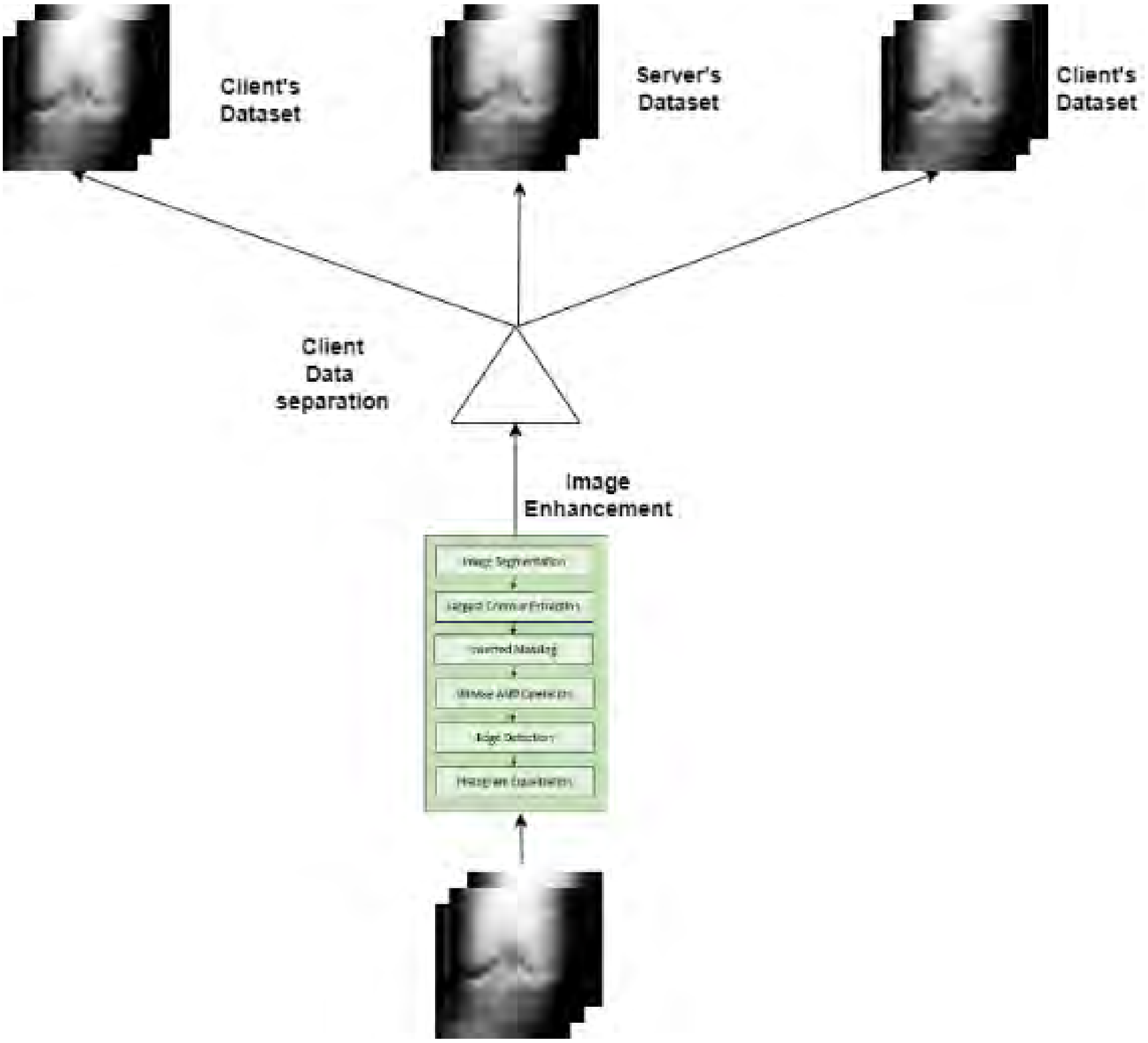


Figure 4.15: Client and Server Dataset Creation of FixMatch-Based Federated Learning Framework

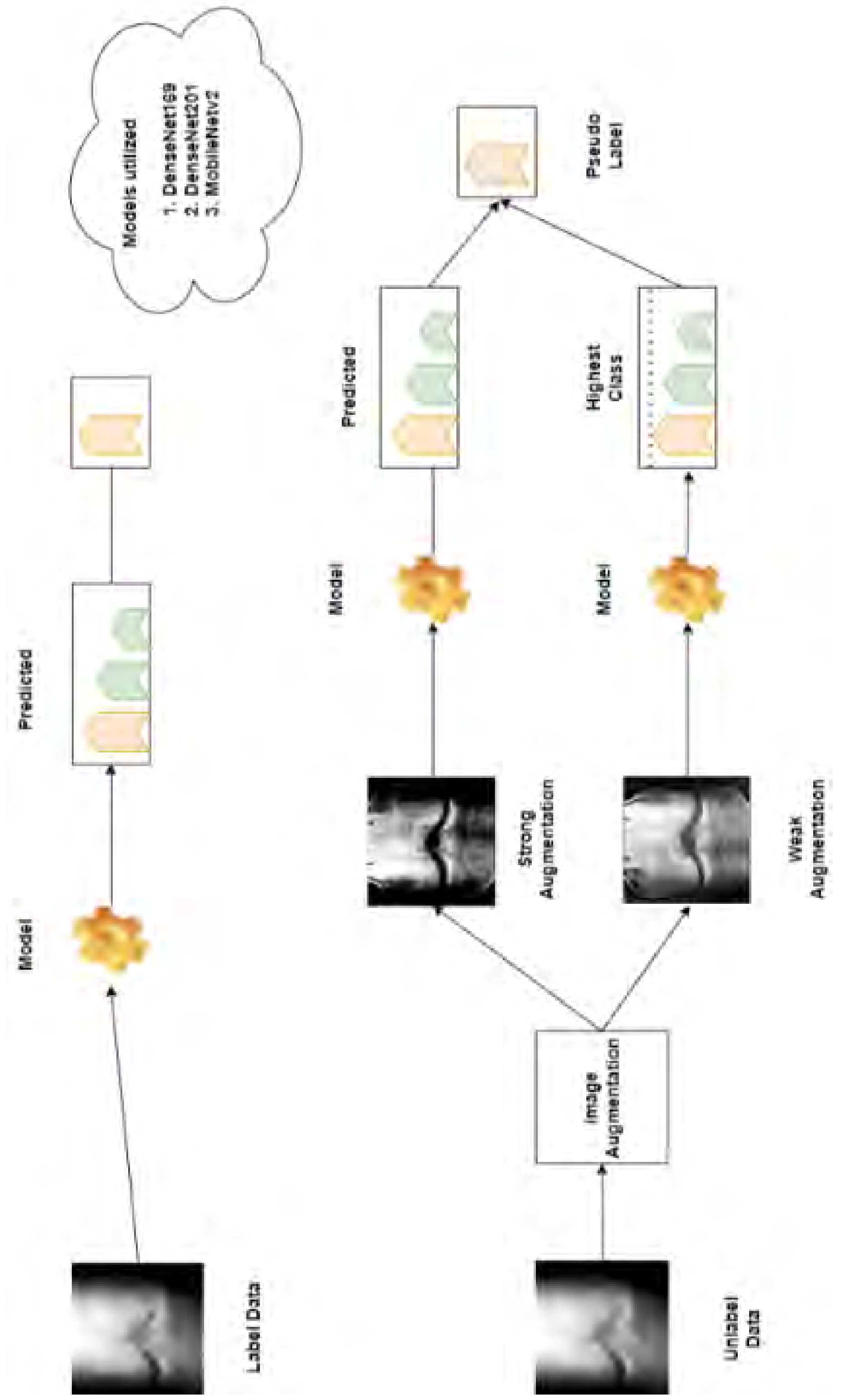


Figure 4.16: Working Procedure of FixMatch

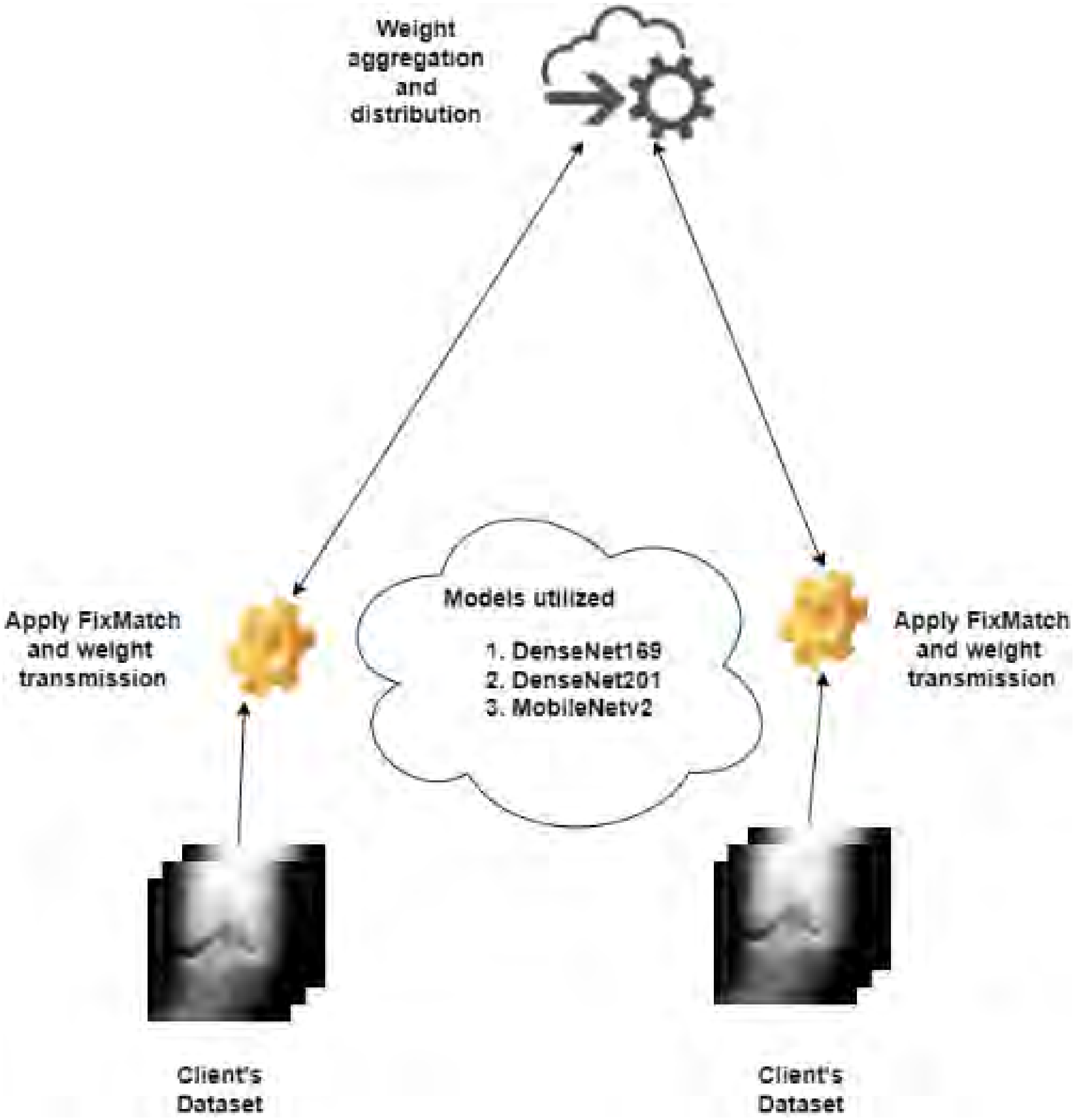

Weight aggregation and distribution
Apply FixMatch and weight transmission
Models utilized
1. DenseNet169
2. DenseNet201
3. MobileNetv2
Apply FixMatch and weight transmission
Client's Dataset
Client's Dataset

Figure 4.17: Working Procedure of FixMatch-Based Federated Learning Framework

Chapter 5

Performance Evaluation

In this portion of the thesis, we shall elucidate the discoveries and draw definitive conclusions from our study. The experimental results are categorized into four categories. The initial stage involves examining the outcomes of the Client-Server-Based Federated Learning Framework, while the subsequent stage will focus on evaluating the results of the pseudo-labeling-based Federated Learning Framework. During the third step, we will examine the FixMatch-Based Federated Learning Framework. In the last part, we will compare the methodologies and analyze the disparities in outcomes between Semi-Supervised Learning (SSL) and Traditional Federated Learning (FL).

5.1 Evaluation Matrices

A variety of performance metrics were used in this study to explain why ML models could perform well with one evaluation metric's measurement while performing not so great with another metric's assessment. In this study, we mainly used Accuracy, Precision, Recall, and F1-Score as performance evaluation metrics.
Here,
TP = True Positive , TN = True Negative
FP = False Positive, FN = False Negative

5.1.1 Accuracy

Accuracy is defined as the total number of accurate predictions divided by the total number of data samples present in the dataset as shown in the equation (5.1)-

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5.1}$$

5.1.2 Precision

The Precision is defined as the total number of accurate positive predictions divided by the total number of positive predictions as shown in the equation (5.2)-

$$Precision = \frac{TP}{TP + FP} \tag{5.2}$$

5.1.3 Recall

The recall is defined as the total number of accurate positive predictions divided by the total number of actual positive predictions as shown in the equation (5.3)-

$$Recall = \frac{TP}{TP + FN} \tag{5.3}$$

5.1.4 F1-Score

F1-Score is the harmonic mean of precision and recall as shown in the equation (5.4)-

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{5.4}$$

5.2 Experimental Setup

We utilized Google Colab Pro Plus with a high RAM configuration. The graphics processing unit (GPU) employed is a Tesla T4. To improve images and separate clients, we utilized a Dell Inspiron 5559 laptop equipped with 8 GB of RAM and an Intel Intel(R) CoreTM i7-6500U CPU @ 2.50 GHz [Cores 2], operating on Microsoft Windows 10 Pro. We utilized Python 3.9 on a Dell Inspiron 5559 laptop as our programming instrument. The outcomes of the model creation and training process were recorded on Wandb[21] for the client-server-based Federated Learning setup. The results of the pseudo-labeling-based Federated Learning and FixMatch-based Federated Learning were logged on CSV.
We utilized Google Drive as a server to store and deliver model weights to clients.

5.3 Hyperparameters

1. **Hyperparameters for Client-Server-Based Federated Learning Framework**

Table 5.1: Overview of Hyperparameters for Client-Server-Based Federated Learning Framework

Hyperparameter	Value
Number of Epochs	10
Initial Learning Rate	1e-4
Batch Size	32
Image Height	224
Image Width	224
Image Depth	3

2. **Hyperparameters for Pseudo-labeling-Based Federated Learning Framework**

Table 5.2: Overview of Hyperparameters for Pseudo-labeling-Based Federated Learning Framework

Hyperparameter	Value
Number of Epochs	10
Initial Learning Rate	1e-4
Batch Size	32
Image Height	224
Image Width	224
Image Depth	3
Pseudo-labeling Threshold	0.70

3. **Hyperparameters for FixMatch-Based Federated Learning**

Table 5.3: Overview of Hyperparameters for FixMatch-Based Federated Learning

Hyperparameter	Value	Description
Number of categories	3	Number of categories or classes in the classification task.
Input shape of image	224, 224, 3	Shape of input images height, width, channels.
mu	2	Size of the unlabeled batch in semi-supervised learning.
lambda u	1	Loss weight to balance supervised and unsupervised losses.
tau	0.8	Weakly augmented threshold controlling augmentation level.
Number of Epochs	10	Number of training epochs.
Learning Rate	1e-4	Learning rate for updating model parameters.
bs lab	2	Batch size for labeled data.
bs unlab	mu * bs lab	Batch size for unlabeled data, calculated as the product of mu and bs lab.
bs total	bs lab + bs unlab	Total batch size including both labeled and unlabeled data batches.

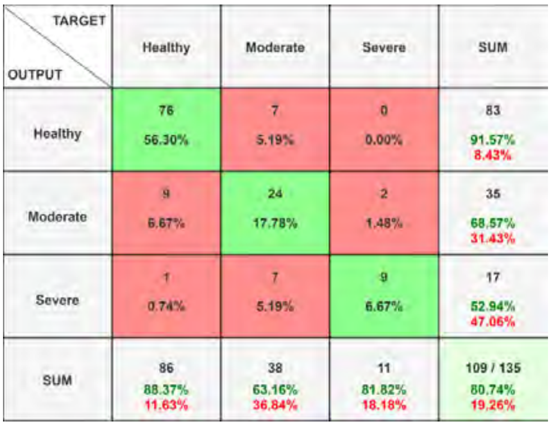
5.4 Performance Analysis: Proposed Method

5.4.1 Pseudo-labeling-Based Federated Learning Framework

For our research, we employed three pre-trained models: DenseNet169, DenseNet201, and MobileNet-v2. In this paragraph, we will analyze the outcomes of the individual model.

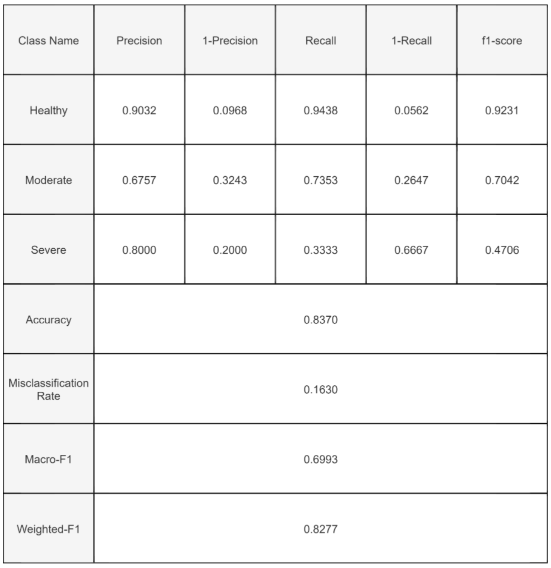
DenseNet169: Initial Server Training

As we have considered that only servers have gold label data, before transmitting the weights to the model, we trained the server model for 20 epochs. After training the model, we got 81% overall accuracy on the test set, 81% weighted average F1-score, 81% weighted average precision, and 81% weighted average recall. Details of the results are given in the confusion matrix 5.1a and classification report 5.4b.



TARGET / OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	76 56.30%	7 5.19%	0 0.00%	83 91.57% 8.43%
Moderate	9 6.67%	24 17.78%	2 1.48%	35 68.57% 31.43%
Severe	1 0.74%	7 5.19%	9 6.67%	17 52.94% 47.06%
SUM	86 88.37% 11.63%	38 63.16% 36.84%	11 81.82% 18.18%	109 / 135 80.74% 19.26%

(a) Confusion Matrix of DenseNet169 in Pseudo-Labeling-Based Federated Learning



Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.9032	0.0968	0.9438	0.0562	0.9231
Moderate	0.6757	0.3243	0.7353	0.2647	0.7042
Severe	0.8000	0.2000	0.3333	0.6667	0.4706
Accuracy	0.8370				
Misclassification Rate	0.1630				
Macro-F1	0.6993				
Weighted-F1	0.8277				

(b) Classification Report of DenseNet169 in Pseudo-Labeling-Based Federated Learning

Figure 5.1: DenseNet169 Performance in Pseudo-Labeling-Based Federated Learning

The server successfully predicted 109 images among 135 images, and most miss-classification was seen in the Moderate class and the overall miss-classification rate was 0.19

DenseNet169: Federated Learning

Upon receiving the model weight from the server, clients embarked on data labeling tasks. In the initial communication round, each client processed its dataset, with Client 1 identifying 672 data points surpassing the prediction confidence threshold which is 70%, while Client 2 successfully labeled 680 data points meeting the same criterion.

Advancing to subsequent communication rounds, Client 1 exhibited improved performance, accurately categorizing 967 data points with high prediction confidence. Meanwhile, Client 2 continued to make strides, successfully identifying 984 data points meeting or exceeding the confidence threshold. The detail of the statistics is presented in Table 5.4

The data indicates a distinct pattern of enhancement in predicted accuracy as the number of communication cycles increases. At the beginning, during round 0, the model's capacity to make accurate predictions was not sufficient to reach a confidence level of 70% for all clients. Nevertheless, when the model underwent repeated

Table 5.4: Number of Pseudo-Labeled Data on Each Communication Round Using DenseNet169

Split	CommR	Healthy	Moderate	Severe
Client 1	0	459	121	92
Client 2	0	467	110	103
Client 1	1	810	123	34
Client 2	1	834	115	35

training using pseudo-labels, there was a noticeable performance improvement. During communication round 1, all clients demonstrated the ability to forecast 50% more data compared to the previous round, thereby highlighting the efficacy of the iterative training method.

It is worth mentioning that although the general accuracy of predictions increased, there was a decrease in the number of predictions in the severe category. This suggests that there are specific areas where the model may be enhanced to better handle cases of severe categorization.

Further analysis of test accuracy demonstrates substantial improvement across all clients. Client 1 had a significant 4% improvement in accuracy from round 0 to round 1, achieving an amazing accuracy rate of 91% in round 1. Client 2, in contrast, showed significant progress, with accuracy increasing by 12% from 85% in round 0 to an outstanding 97% in round 1. The server's accuracy had a significant 14% boost, going from 70% in round 1 to 84% in round 1.

The findings highlight the effectiveness of the training technique used, as seen by the steady enhancement in predictive performance and accuracy measures for both clients and the server. These observations not only confirm the iterative training strategy but also offer useful suggestions for improving the model's performance in future rounds.

Table 5.5: Test Results on DenseNet169 in Pseudo-Labeling-Based Federated Learning

Model	CommR	Client1	Client2	Server
DenseNet169	0	0.91	0.85	0.70
	1	0.96	0.97	0.84

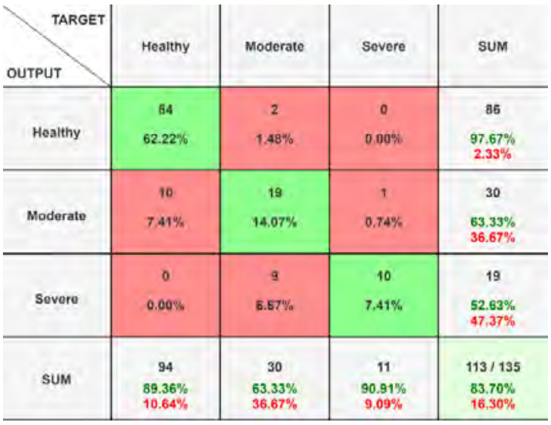
A detailed result can be found in The Table 5.6, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.6: Detailed Test Results of Each Communication Round Using DenseNet169 in Pseudo-Labeling-Based Federated Learning

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
DenseNet169	Client1	0	0.91	0.91	0.91	0.91
	Client2	0	0.85	0.87	0.85	0.86
	Server	0	0.70	0.77	0.77	0.72
	Client1	1	0.96	0.96	0.96	0.96
	Client2	1	0.97	0.97	0.97	0.97
	Server	1	**0.84**	**0.84**	**0.84**	**0.83**

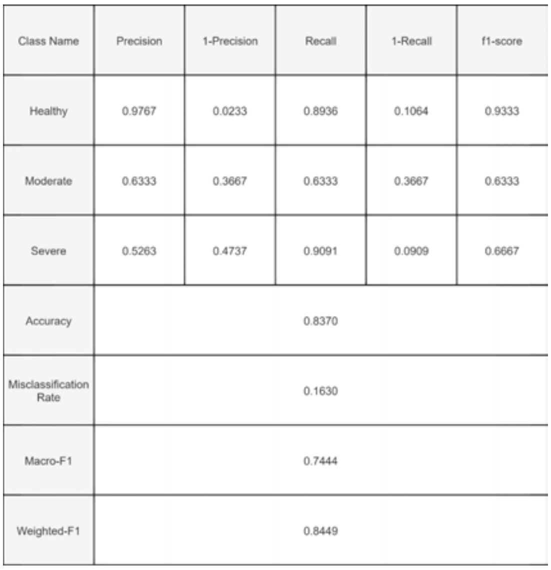
DenseNet201: Initial Server Training

As we have considered that only servers have gold label data, before transmitting the weights to the model, we trained the server model for 20 epochs. After training the model, we got 86% overall accuracy on the test set, 86% weighted average F1-score, 87% weighted average precision, and 86% weighted average recall. Details of the results are given in the confusion matrix 5.2a and classification report 5.2b. The server successfully predicted 116 images among 135 images, and most miss-classification were seen in the Moderate class the overall miss-classification rate was 0.14.



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	84 62.22%	2 1.48%	0 0.00%	86 97.67% 2.33%
Moderate	10 7.41%	19 14.07%	1 0.74%	30 63.33% 36.67%
Severe	0 0.00%	9 6.67%	10 7.41%	19 52.63% 47.37%
SUM	94 89.36% 10.64%	30 63.33% 36.67%	11 90.91% 9.09%	113 / 135 83.70% 16.30%

(a) Confusion Matrix of DenseNet201 in Pseudo-Labeling-Based Federated Learning



Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.9767	0.0233	0.8936	0.1064	0.9333
Moderate	0.6333	0.3667	0.6333	0.3667	0.6333
Severe	0.5263	0.4737	0.9091	0.0909	0.6667
Accuracy	0.8370				
Misclassification Rate	0.1630				
Macro-F1	0.7444				
Weighted-F1	0.8449				

(b) Classification Report of DenseNet201 in Pseudo-Labeling-Based Federated Learning

Figure 5.2: DenseNet201 Performance in Pseudo-Labeling-Based Federated Learning

DenseNet201: Federated Learning

After obtaining the weight of the model from the server, the clients began their data labeling jobs. During the initial communication session, each client analyzed their dataset. Client 1 identified 809 data points that exceeded the prediction confidence level of 70%. Also, Client 2 correctly labeled 832 images which met the same criteria. Client 1 showed enhanced performance in communication round 1, successfully classifying 836 images. Meanwhile, Client 2 made significant progress by successfully identifying 873 images with the progress of the communication round. The detailed statistics are presented in Table 5.7

Table 5.7: Number of Pseudo-Labeled Data on Each Communication Round Using DenseNet201

Split	CommR	Healthy	Moderate	Severe
Client 1	0	566	209	34
Client 2	0	591	209	32
Client 1	1	534	224	78
Client 2	1	579	218	76

According to Table 5.8, in communication round 0, the model demonstrated a 70% degree of confidence in accurately predicting all of the data. Nevertheless, the degree of confidence rapidly increased when the model underwent training using pseudo-labels. In the first round of communication, all clients demonstrated the ability to forecast 50% additional data in the severe category compared to the prior round.

Table 5.8: Test Results on DenseNet201 in Pseudo-Labeling-Based Federated Learning

Model	CommR	Client1	Client2	Server
DenseNet201	0	0.90	0.90	0.82
	1	0.93	0.92	0.84

In addition, the test accuracy of each server and client may be characterized in the following manner: Client 1 attained a starting accuracy of 90% in communication round 0, which then improved by an additional 4% in communication round 1. However, Client 2 saw a notable improvement in accuracy, with a 92% rise in communication round 1 compared to its original accuracy of 90% in communication round 0. The server exhibited enhanced performance, with its precision rising by 2% from 82% in round 0 to 84% in round 1.

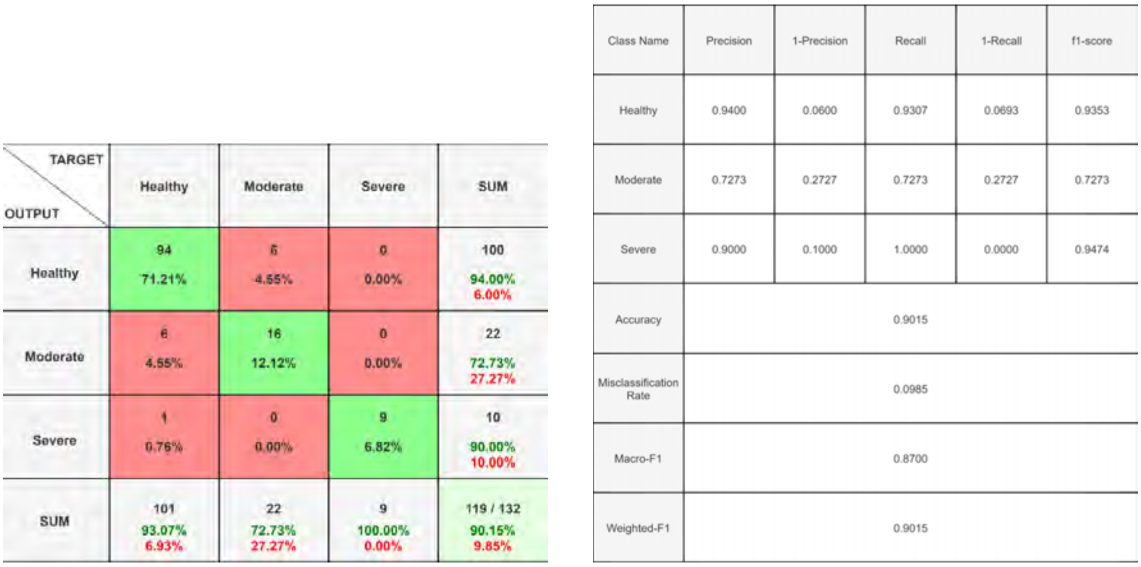
A detailed result can be found in the Table 5.9, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.9: Detailed Test Results of Each Communication Round Using DenseNet201 on Pseudo-Labeling-Based Federated Learning

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
	Client1	0	0.90	0.90	0.90	0.90
	Client2	0	0.90	0.81	0.90	0.90
	Server	0	0.82	0.83	0.82	0.82
	Client1	1	0.93	0.93	0.93	0.93
	Client2	1	0.92	0.92	0.92	0.92
DenseNet201	Server	1	**0.84**	**0.84**	**0.84**	**0.84**

MobileNet V2: Initial Server Training

As we have considered that only servers have gold label data, before transmitting the weights to the model, we trained the server model for 20 epochs. After training the model, we got 84% overall accuracy on the test set, 84% weighted average F1-score, 83% weighted average precision, and 84% weighted average recall. Details of the results are given in the confusion matrix 5.3a and classification report 5.3b.



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	94 71.21%	6 4.55%	0 0.00%	100 94.00% 6.00%
Moderate	6 4.55%	16 12.12%	0 0.00%	22 72.73% 27.27%
Severe	1 0.76%	0 0.00%	9 6.82%	10 90.00% 10.00%
SUM	101 93.07% 6.93%	22 72.73% 27.27%	9 100.00% 0.00%	119 / 132 90.15% 9.85%

(a) Confusion Matrix of MobileNetV2 in Pseudo-Labeling-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.9400	0.0600	0.9307	0.0693	0.9353
Moderate	0.7273	0.2727	0.7273	0.2727	0.7273
Severe	0.9000	0.1000	1.0000	0.0000	0.9474
Accuracy	0.9015				
Misclassification Rate	0.0985				
Macro-F1	0.8700				
Weighted-F1	0.9015				

(b) Classification Report of MobileNetV2 in Pseudo-Labeling-Based Federated Learning

Figure 5.3: MobileNetV2 Performance in Pseudo-Labeling-Based Federated Learning

Our server successfully predicted 114 images among 135 images, and most miss-classification was seen in the Moderate class the overall miss-classification rate was 0.15

MobileNet V2: Federated Learning

Once the client was provided with the model weight by the server, it immediately began the process of labeling the data. When it comes to communication round 0, client 1 has the ability to pass the threshold confidence with 952 data points. The second client, which has 969 data points, is able to satisfy the requirements.
There were 966 data points that were able to pass the criterion, which was 70%, during the first communication session with client 1. In addition, 1013 photos were labeled for the second client. The detailed statistics is presented in Table 5.10

Table 5.10: Number of Pseudo-Labeled Data on Each Communication Round Using MobileNetV2

Split	CommR	Healthy	Moderate	Severe
Client 1	0	725	150	77
Client 2	0	591	209	79
Client 1	1	661	219	86
Client 2	1	711	214	88

What we can notice from the table 5.11 is that in communication round 0, the model was unable to predict all data with 70% confidence, but it increased gradually when it trained on pseudo-labels, and in communication round 1, all clients were able to predict more data than in every class than the previous round.
On the other hand, the test accuracy of each server and client follows this client 1 got an overall 94% accuracy on communication round 0 which decreased by 1% more in communication round 1, whereas client 2 increased in accuracy in communication round 1 by 92% where in communication round 0 it was 86% sever also shows and increment in the round to by 4% from 80% it rose to 84%.

Table 5.11: Test Results on MobileNetV2 in Pseudo-Labeling-Based Federated Learning

Model	CommR	Client1	Client2	Server
MobileNet V2	0	0.94	0.86	0.80
	1	0.93	0.92	0.84

A detailed result can be found in Table 5.12, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.12: Detailed Test Results of Each Communication Round Using MobileNetV2 on Pseudo-Labeling-Based Federated Learning

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
MobileNet V2	Client1	0	0.94	0.94	0.94	0.94
	Client2	0	0.86	0.89	0.86	0.86
	Server	0	0.80	0.81	0.80	0.80
	Client1	1	0.93	0.93	0.93	0.93
	Client2	1	0.92	0.92	0.92	0.92
	Server	1	**0.88**	**0.88**	**0.88**	**0.88**

Overall Comparison

If we compare all three models in pseudo-labeling-based federated learning, we will see that in the case of the client's data labeling, MobilenetV2 outperforms DenseNet169 and DenseNet201. In round 1 where every model performed better than the previous round MobilenetV2 was able to predict 1013 images with single class prediction confidence 70% where the value was 984 and 874 for DenseNet169 and DenseNet201 for client two even in client 1 MobilenetV2 performed very well

Table 5.13: Number of Pseudo-Labeled Data on Communication Round-1 Using All Pre-trained Models

Model Name	Split	Healthy	Moderate	Severe
DenseNet169	Client 1	810	123	34
	Client 2	834	115	35
DenseNet201	Client 1	534	224	78
	Client 2	579	218	76
MobileNetV2	Client 1	661	219	86
	Client 2	711	214	**88**

We can see from the details in Table 5.13 that DenseNet201 was better than DenseNet169 in the case of predicting the severe class, but in the comparison of moderate and severe classes, where the data amount was small, mobile net performed better than both models in pseudo-labeling.

Now if we look at the overall model performance in communication round 1 in Table 5.14, after aggregating the weights received from the clients, Both DenseNet models performed similarly in the test portion of the data, while MobileNet v2 performed much better in comparison to the two models.

Considering computational efficiency, while DenseNet169 and DenseNet201 exhibit similar computation times, MobileNetV2 requires slightly more time. However, this marginal increase in computation time is justified by MobileNetV2's superior performance and accuracy.

If we look at the confusion matrix 5.4a, 5.5a and 5.6a and the classification reports 5.4b, 5.5b and 5.6b of the test portion, then we can notice the difference between

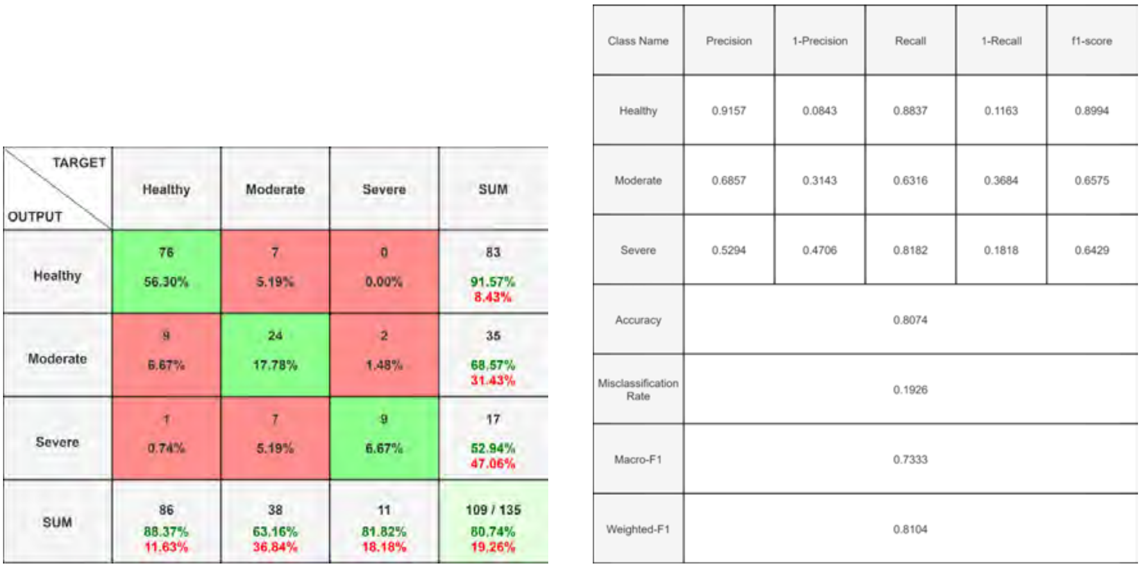
Table 5.14: Detailed Test Results on Different Models of Pseudo-Labeling-Based Federated Learning

Model	Accuracy	Precision	Recall	F1-Score
DenseNet169	0.84	0.84	0.84	0.83
DenseNet201	0.84	0.84	0.84	0.84
MobileNet v2	**0.88**	**0.88**	**0.88**	**0.88**

Table 5.15: Comparison of Computation Time in Different Models in Pseudo-Labeling-Based Federated Learning

Model	Computation Time(min)
DenseNet169	**18.04**
DenseNet201	20.20
MobileNet v2	18.34

all three models more clearly. The miss-classification rate is 0.09 for MobileNetV2, while for DenseNet169 and DenseNet201 it is 0.16. The correct classification rate is also high for all three classes of MobileNetV2.



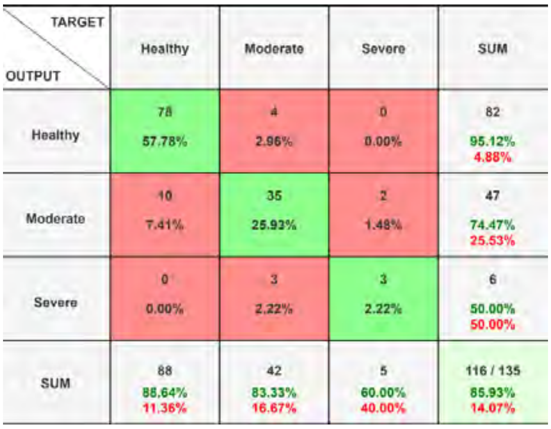
TARGET / OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	76 56.30%	7 5.19%	0 0.00%	83 91.57% 8.43%
Moderate	9 6.67%	24 17.78%	2 1.48%	35 68.57% 31.43%
Severe	1 0.74%	7 5.19%	9 6.67%	17 52.94% 47.06%
SUM	86 88.37% 11.63%	38 63.16% 36.84%	11 81.82% 18.18%	109 / 135 80.74% 19.26%

(a) Confusion Matrix of DenseNet169 in Pseudo-Labeling-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.9157	0.0843	0.8837	0.1163	0.8994
Moderate	0.6857	0.3143	0.6316	0.3684	0.6575
Severe	0.5294	0.4706	0.8182	0.1818	0.6429
Accuracy	0.8074				
Misclassification Rate	0.1926				
Macro-F1	0.7333				
Weighted-F1	0.8104				

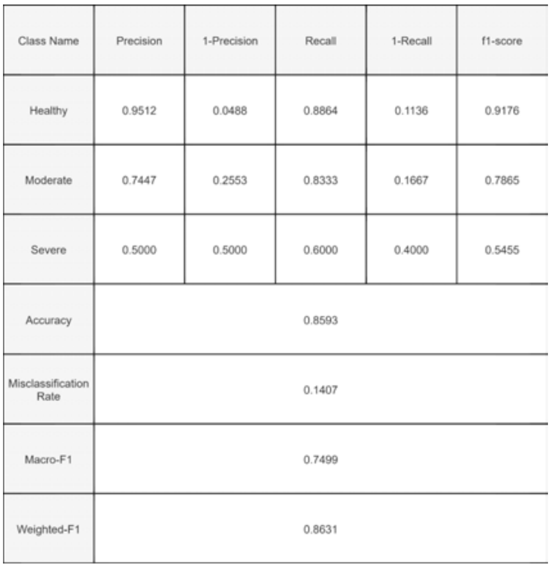
(b) Classification Report of DenseNet169 in Pseudo-Labeling-Based Federated Learning

Figure 5.4: DenseNet169 Performance in Pseudo-Labeling-Based Federated Learning



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	78 57.78%	4 2.96%	0 0.00%	82 95.12% 4.88%
Moderate	10 7.41%	35 25.93%	2 1.48%	47 74.47% 25.53%
Severe	0 0.00%	3 2.22%	3 2.22%	6 50.00% 50.00%
SUM	88 88.64% 11.36%	42 83.33% 16.67%	5 60.00% 40.00%	116 / 135 85.93% 14.07%

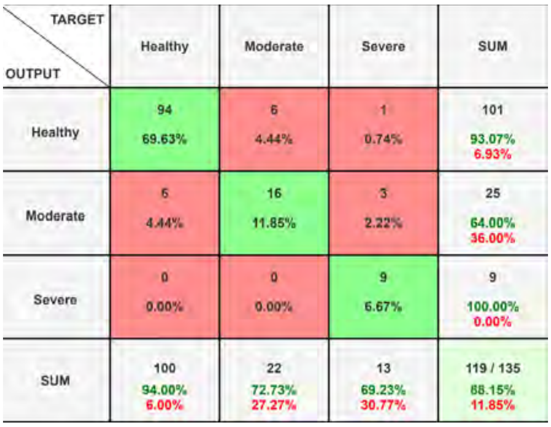
(a) Confusion Matrix of DenseNet201 in Pseudo-Labeling-Based Federated Learning



Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.9512	0.0488	0.8864	0.1136	0.9176
Moderate	0.7447	0.2553	0.8333	0.1667	0.7865
Severe	0.5000	0.5000	0.6000	0.4000	0.5455
Accuracy			0.8593		
Misclassification Rate			0.1407		
Macro-F1			0.7499		
Weighted-F1			0.8631		

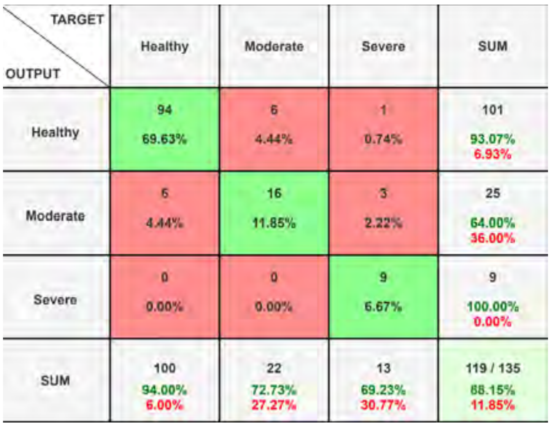
(b) Classification Report of DenseNet201 in Pseudo-Labeling-Based Federated Learning

Figure 5.5: DenseNet201 Performance in Pseudo-Labeling-Based Federated Learning



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	94 69.63%	6 4.44%	1 0.74%	101 93.07% 6.93%
Moderate	6 4.44%	16 11.85%	3 2.22%	25 64.00% 36.00%
Severe	0 0.00%	0 0.00%	9 6.67%	9 100.00% 0.00%
SUM	100 94.00% 6.00%	22 72.73% 27.27%	13 69.23% 30.77%	119 / 135 88.15% 11.85%

(a) Confusion Matrix of MobileNetV2 in Pseudo-Labeling-Based Federated Learning



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	94 69.63%	6 4.44%	1 0.74%	101 93.07% 6.93%
Moderate	6 4.44%	16 11.85%	3 2.22%	25 64.00% 36.00%
Severe	0 0.00%	0 0.00%	9 6.67%	9 100.00% 0.00%
SUM	100 94.00% 6.00%	22 72.73% 27.27%	13 69.23% 30.77%	119 / 135 88.15% 11.85%

(b) Classification Report of MobileNetV2 in Pseudo-Labeling-Based Federated Learning

Figure 5.6: MobileNetV2 Performance in Pseudo-Labeling-Based Federated Learning

5.5 Performance Analysis: Comparative Methods

5.5.1 Client-Server-Based Federated Learning Framework

For our research, we employed two pre-trained models: DenseNet169 and MobileNet-v2. In this paragraph, we will analyze the outcomes of the individual model.
The table displays the accuracy of the two pre-trained models used in the test dataset, as well as the accuracy of their information on both the client and server sides. Table 5.16 clearly demonstrates that DenseNet169 outperforms the other models discussed before. All the models illustrating the impact of federated learning exhibited enhanced accuracy throughout the communication round.

Table 5.16: Test Results on Pre-trained Models in Client-Server-Based Federated Learning

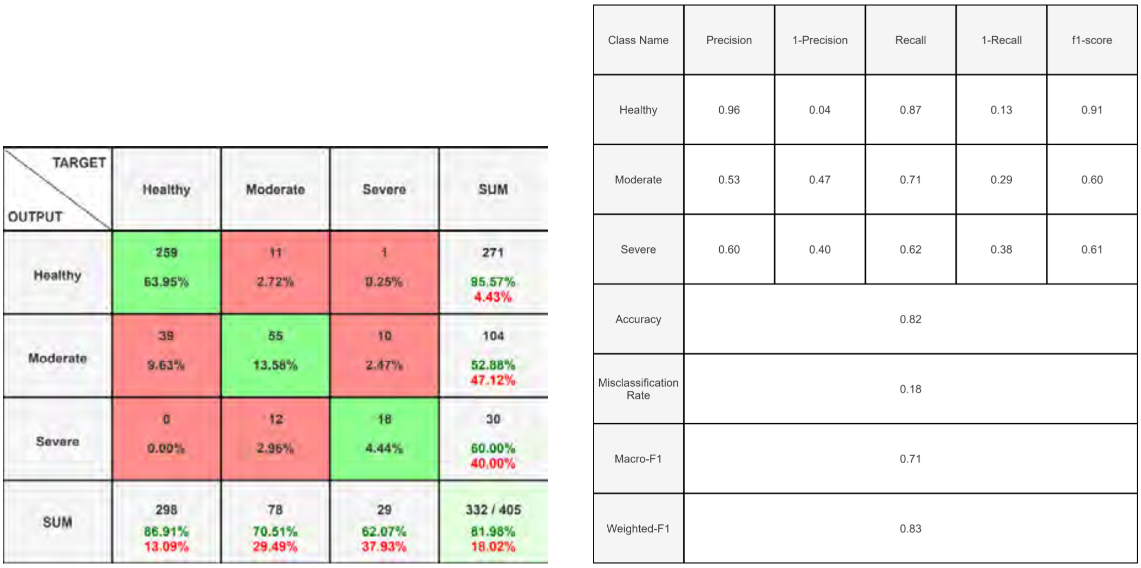
Model	CommR	Client1	Client2	Server
DenseNet169	0	0.85165	0.87978	0.69966
	1	0.86183	0.9071	0.82512
	2	0.86264	0.90164	0.81773
MobileNet v2	0	0.83516	0.78142	0.44828
	1	0.87636	0.8306	0.76847
	2	0.85714	0.86339	0.81034

Table 5.17 shows the test results for several models. We can see that the DenseNet169 model, which has an accuracy rate of over 0.82, performs the best. The DenseNet169 model has the highest F1 score, which is 0.81. The MobileNet-v2 model's F1-score, which is 0.80, is satisfactory. The accuracy of the MobileNet-v2 is 0.81. Both models performed admirably with essentially the same outcomes across all evaluation matrices. DenseNet's ability to reuse features from different layers might be advantageous in capturing global patterns across diverse datasets. Furthermore, DenseNet169 is a relatively deeper model compared to MobileNet-v2, which could allow it to learn more intricate representations.

Table 5.17: Detailed Test Results on Different Models in Client-Server-Based Federated Learning

Model	Accuracy	Precision	Recall	F1-Score
DenseNet169	0.82	0.81	0.82	0.81
MobileNet v2	0.81	0.81	0.81	0.80

In Figure 5.7a,5.7b,5.8a and 5.8b, we can see the confusion matrices and get the details about the classifications. It is seen that the DenseNet169 model classifies 259 healthy graded Knee joints correctly while miss-classifying 12 data. MobileNet-v2 classifies the healthy data which is 256 and misclassifies 15 data into other classes. The class that is mostly misclassified by all models is the Moderate class. Both models classify the moderate-graded KOA as a Healthy knee. A total of 50 data are misclassified by the MobileNet-v2 model followed by 49 miss-classification by the DenseNet169 model. These models perform nearly the same while classifying Severe Knee OA predicting 19 and 18 data correctly respectively.



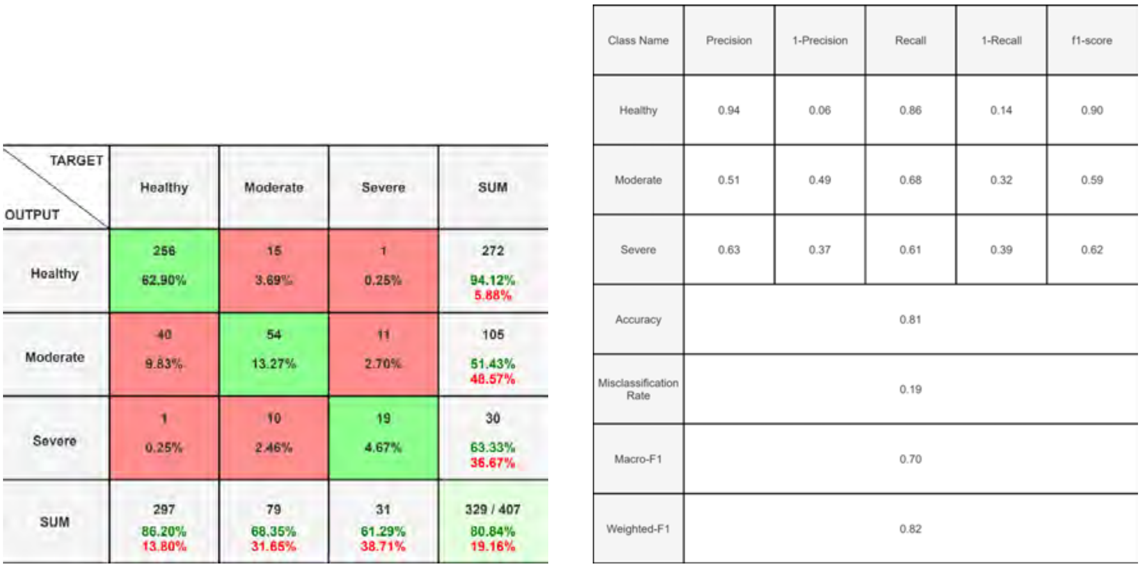
TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	259 63.95%	11 2.72%	1 0.25%	271 95.57% 4.43%
Moderate	39 9.63%	55 13.58%	10 2.47%	104 52.88% 47.12%
Severe	0 0.00%	12 2.96%	18 4.44%	30 60.00% 40.00%
SUM	298 86.91% 13.09%	78 70.51% 29.49%	29 62.07% 37.93%	332 / 405 81.98% 18.02%

(a) Confusion Matrix of DenseNet169 in Client-Server-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.96	0.04	0.87	0.13	0.91
Moderate	0.53	0.47	0.71	0.29	0.60
Severe	0.60	0.40	0.62	0.38	0.61
Accuracy	0.82				
Misclassification Rate	0.18				
Macro-F1	0.71				
Weighted-F1	0.83				

(b) Classification Report of DenseNet169 in Client-Server-Based Federated Learning

Figure 5.7: DenseNet169 Performance in Client-Server-Based Federated Learning



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	256 62.90%	15 3.69%	1 0.25%	272 94.12% 5.88%
Moderate	40 9.83%	54 13.27%	11 2.70%	105 51.43% 48.57%
Severe	1 0.25%	10 2.46%	19 4.67%	30 63.33% 36.67%
SUM	297 86.20% 13.80%	79 68.35% 31.65%	31 61.29% 38.71%	329 / 407 80.84% 19.16%

(a) Confusion Matrix of MobileNetV2 in Client-Server-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.94	0.06	0.86	0.14	0.90
Moderate	0.51	0.49	0.68	0.32	0.59
Severe	0.63	0.37	0.61	0.39	0.62
Accuracy	0.81				
Misclassification Rate	0.19				
Macro-F1	0.70				
Weighted-F1	0.82				

(b) Classification Report of MobileNetV2 in Client-Server-Based Federated Learning

Figure 5.8: MobileNetV2 Performance in Client-Server-Based Federated Learning

Table 5.18 displays the computational variance between the two models. The table shows that even though the DenseNet169 model is the most accurate, it still requires the longest computation times. In contrast, the MobileNet v2 model runs roughly 49.716 minutes faster than the DenseNet169 model while exhibiting almost the same accuracy.

Table 5.18: Comparison of Computation Time in Different Models in Client-Server-Based Federated Learning

Model	Computation Time(min)
DenseNet169	59.517
MobileNet v2	**49.716**

5.5.2 FixMatch-Based Federated Learning Framework

DenseNet169

In both clients 1 and 2, there was a consistent improvement in accuracy across communication rounds. Client 1 started at 82% accuracy in round 0 and maintained the same accuracy in round 1. Client 2 improved from 80% in round 0 to 83% in round 1. The server showed a significant improvement from 70% accuracy in round 0 to 88% in round 1. These improvements were reflected in the weighted average precision, recall, and F1-score for each entity, demonstrating the effectiveness of the iterative communication process in refining the model's performance. The detailed statistics is presented in Table 5.19

Table 5.19: Test Results on DenseNet169 in FixMatch-Based Federated Learning Framework

Model	CommR	Client1	Client2	Server
DenseNet169	0	0.82	0.80	0.70
	1	0.82	0.83	0.88

A detailed result can be found in the Table 5.20, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.20: Detailed Test Results of Each Communication Round Using DenseNet169 on FixMatch-Based Federated Learning Framework

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
DenseNet169	Client1	0	0.82	0.82	0.82	0.79
	Client2	0	0.80	0.79	0.80	0.78
	Server	0	0.70	0.68	0.70	0.68
	Client1	1	0.82	0.83	0.82	0.80
	Client2	1	0.83	0.82	0.83	0.81
	Server	1	0.88	0.84	0.88	0.86

DenseNet201

All throughout the communication rounds, both client 1 and client 2 displayed an increased level of accuracy. In the first round, Client 1's accuracy increased from 80% in the first round to 84% in the first round. The accuracy rate of Client 2 was consistent across both rounds, coming in at 86%. on the other hand, the server's performance, stayed unchanged across both rounds, with an accuracy rate of 68%. Although there was a slight increase in the server's accuracy, recall, and F1 score from round 0 to round 1, these metrics continued to be much lower than those of the clients. In light of this, it may be deduced that the performance of the server has space for continued improvement. The detailed statistics is presented in Table 5.21

Table 5.21: Test Results on DenseNet201 in FixMatch-Based Federated Learning

Model	CommR	Client1	Client2	Server
DenseNet201	0	0.80	0.86	0.68
	1	0.84	0.84	0.80

A detailed result can be found in this table 5.22, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.22: Detailed Test Results of Each Communication Round Using DenseNet201 on FixMatch-Based Federated Learning

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
DenseNet201	Client1	0	0.80	0.80	0.80	0.79
	Client2	0	0.86	0.86	0.86	0.84
	Server	0	0.68	0.46	0.68	0.55
	Client1	1	0.84	0.85	0.84	0.83
	Client2	1	0.84	0.85	0.84	0.83
	Server	1	0.80	0.74	0.80	0.77

MobileNetV2

In both clients 1 and 2, there was an improvement in accuracy across communication rounds. Client 1 started at 83% accuracy in round 0 and increased to 84% in round 1. Client 2 also showed improvement from 81% accuracy in round 0 to 87% in round 1. However, the server's performance remained stable with 76% accuracy in both rounds. While there were improvements in the server's precision, recall, and F1-score from round 0 to round 1, they were still lower compared to the clients, indicating potential for further enhancement in its performance. The detailed statistics are presented in Table 5.23

Table 5.23: Test Results on MobileNetV2 in FixMatch-Based Federated Learning

Model	CommR	Client1	Client2	Server
MobileNet V2	0	0.83	0.81	0.76
	1	0.84	0.87	0.8

A detailed result can be found in Table 5.24, where we can see that in all metrics, we got an impressive result in both clients 1 and 2, and with each communication round, the result got even better.

Table 5.24: Detailed Test Results of Each Communication Round Using MobileNetV2 on FixMatch-Based Federated Learning

Model	Client	CommR	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
MobileNet V2	Client1	0	0.83	0.83	0.83	0.81
	Client2	0	0.81	0.82	0.81	0.80
	Server	0	0.76	0.65	0.76	0.67
	Client1	1	0.84	0.84	0.84	0.82
	Client2	1	0.87	0.87	0.87	0.85
	Server	1	0.87	0.86	0.87	0.76

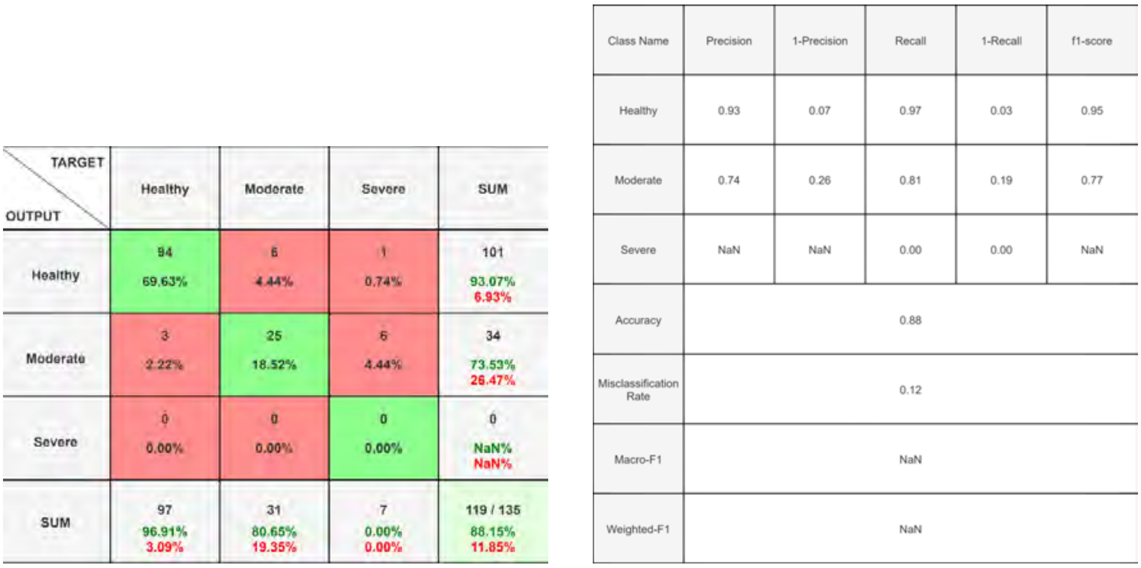
Overall Comparison

Upon evaluating the total performance of the models in communication round 1 and combining the weights obtained from the customers, it is clear that both DenseNet models exhibited similar levels of performance in the test dataset. MobileNet v2 exhibited markedly superior performance levels in comparison to the DenseNet models. Furthermore, MobileNet v2 demonstrated a commendable accuracy of 0.87, indicating its proficiency in accurately classifying KAO pictures in the given test dataset. Furthermore, it showcased a remarkable precision of 0.86 and a recall of 0.87, indicating its efficacy in accurately identifying positive examples and retrieving all pertinent instances from the dataset, respectively. The F1-Score of 0.76 provides solid evidence that the model's performance is equally accurate and comprehensive. MobileNet v2 has exhibited significantly improved performance compared to DenseNet169 and DenseNet201, indicating that it is a more suitable choice for the particular task and dataset. The detailed statistics are presented in Table 5.25

Table 5.25: Detailed Test Results on Different Models on FixMatch-Based Federated Learning

Model	Accuracy	Precision	Recall	F1-Score
DenseNet169	0.88	0.84	0.88	0.86
DenseNet201	0.80	0.74	0.80	0.77
MobileNet v2	0.87	0.86	0.87	0.76

If we look at the confusion matrix 5.9a,5.10a and 5.11a and the classification report 5.9b,5.10b and 5.11b of the test portion, then we can notice the difference between all three models more clearly. The DenseNet169 and DenseNet201 were able to identify the classes Healthy and Moderate but it was completely unable to detect a single image of class severe in the test dataset, whereas MobileNetV2 was able to classify only a single image of class severe



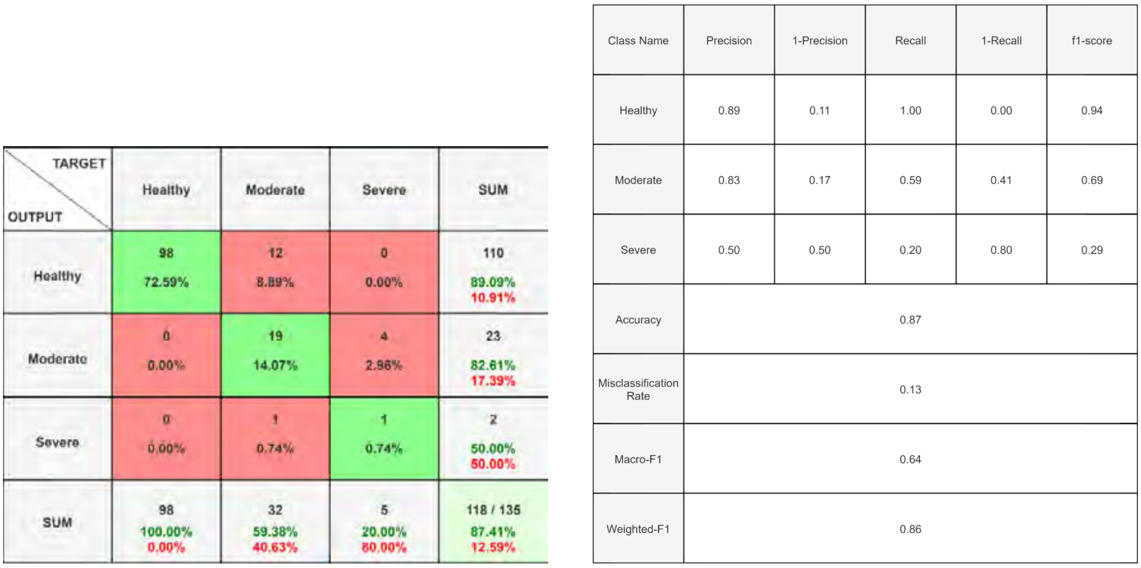
TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	94 69.63%	6 4.44%	1 0.74%	101 93.07% 6.93%
Moderate	3 2.22%	25 18.52%	6 4.44%	34 73.53% 26.47%
Severe	0 0.00%	0 0.00%	0 0.00%	0 NaN% NaN%
SUM	97 96.91% 3.09%	31 80.65% 19.35%	7 0.00% 0.00%	119 / 135 88.15% 11.85%

(a) Confusion Matrix of DenseNet169 in FixMatch-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.93	0.07	0.97	0.03	0.95
Moderate	0.74	0.26	0.81	0.19	0.77
Severe	NaN	NaN	0.00	0.00	NaN
Accuracy	0.88				
Misclassification Rate	0.12				
Macro-F1	NaN				
Weighted-F1	NaN				

(b) Classification Report of DenseNet169 in FixMatch-Based Federated Learning

Figure 5.9: DenseNet169 Performance in FixMatch-Based Federated Learning



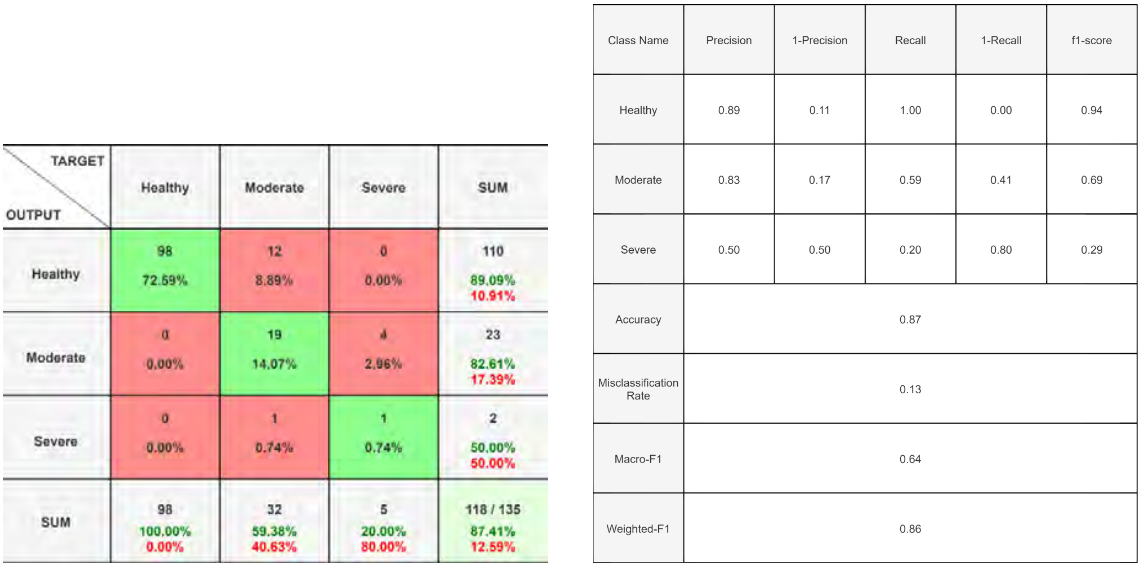
TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	98 72.59%	12 8.89%	0 0.00%	110 89.09% 10.91%
Moderate	0 0.00%	19 14.07%	4 2.96%	23 82.61% 17.39%
Severe	0 0.00%	1 0.74%	1 0.74%	2 50.00% 50.00%
SUM	98 100.00% 0.00%	32 59.38% 40.63%	5 20.00% 80.00%	118 / 135 87.41% 12.59%

(a) Confusion Matrix of DenseNet201 in FixMatch-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.89	0.11	1.00	0.00	0.94
Moderate	0.83	0.17	0.59	0.41	0.69
Severe	0.50	0.50	0.20	0.80	0.29
Accuracy	0.87				
Misclassification Rate	0.13				
Macro-F1	0.64				
Weighted-F1	0.86				

(b) Classification Report of DenseNet201 in FixMatch-Based Federated Learning

Figure 5.10: DenseNet201 Performance in FixMatch-Based Federated Learning



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	98 72.59%	12 8.89%	0 0.00%	110 89.09% 10.91%
Moderate	0 0.00%	19 14.07%	4 2.96%	23 82.61% 17.39%
Severe	0 0.00%	1 0.74%	1 0.74%	2 50.00% 50.00%
SUM	98 100.00% 0.00%	32 59.38% 40.63%	5 20.00% 80.00%	118 / 135 87.41% 12.59%

(a) Confusion Matrix of MobileNetV2 in FixMatch-Based Federated Learning

Class Name	Precision	1-Precision	Recall	1-Recall	f1-score
Healthy	0.89	0.11	1.00	0.00	0.94
Moderate	0.83	0.17	0.59	0.41	0.69
Severe	0.50	0.50	0.20	0.80	0.29
Accuracy	0.87				
Misclassification Rate	0.13				
Macro-F1	0.64				
Weighted-F1	0.86				

(b) Classification Report of MobileNetV2 in FixMatch-Based Federated Learning

Figure 5.11: MobileNetV2 Performance in FixMatch-Based Federated Learning

Table 5.26: Comparison of Computation Time in Different Models in FixMatch-Based Federated Learning

Model	Computation Time(min)
DenseNet169	94.21
DenseNet201	108.68
MobileNet v2	**31.54**

From the table, it can be clearly stated that DenseNet201 took the longest time to train both clients, due to the model depth it took this time. Though DenseNet201 DenseNet169 took longer than MobileNetV2, nearly one-third of DenseNet169, MobileNetV2 performed best among these three models.

5.6 Discussion

When we look at the three model models of a client-server-based federated learning framework, then pseudo-labeling-based federated learning framework, and the FixMatch-Based Federated Learning Framework, we can see that for client-server-based Federated Learning Framework, Densenet169 was the best performer. When it comes to the pseudo-labeling-based Federated Learning Framework, MobileNetV2 is the winner, and in the FixMatch-Based Federated Learning Framework, again, MobileNetV2 outperforms other models.

When comparing the evaluation metrics of these three models in the Table 5.27, we can see that MobileNetV2 in the pseudo-labeling-based Federated Learning Framework outperforms the other two models in different frameworks in every metric. It has the highest overall accuracy (weighted average). Precision, weighted average Recall, and weighted average F1-Score

Table 5.27: Detailed Test Result Comparison of Best-Performing Models in All Three Frameworks

Framework	Model	Accuracy	Weighted Average Precision	Weighted Average Recall	Weighted Average F1-Score
CSFL	Densenet169	0.82	0.81	0.82	0.81
PLFL	MobileNet V2	0.88	0.88	0.88	0.88
FSSFL	MobileNet V2	0.87	0.86	0.87	0.76

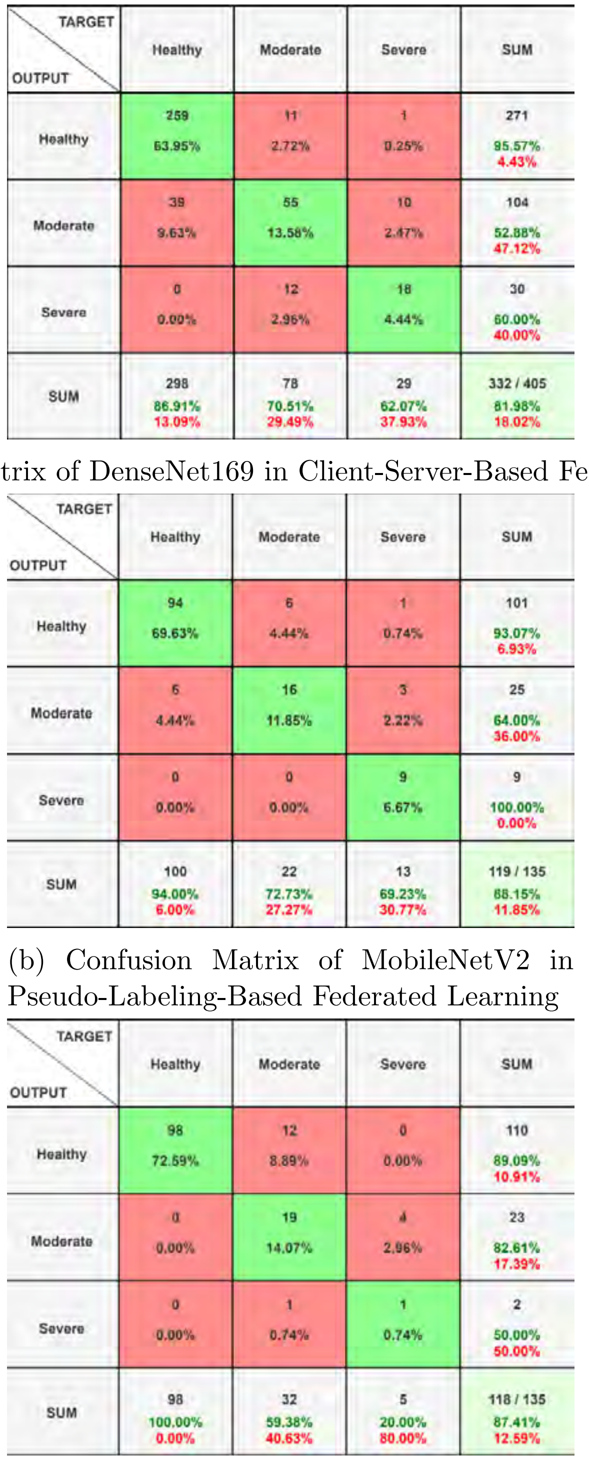
When we look at the training times of the best three models displayed in 5.28, we can see that MobileNetV2 in the pseudo-labeling-based Federated Learning Framework took the longest time to complete label generation and model training, which is 3.2 times lower than DenseNet169, which is trained using the traditional Federated Learning Framework model, and 1.7 times lower than MobileNet v2 in the FixMatch-Based Federated Learning Framework.

Considering the predictions by each model, we can see that DenseNet169 provided an 86.91% correct result on healthy images, a 70.51% correct prediction for the moderate class, and a 62.07% correct prediction for the severe class. MobileNetV2

Table 5.28: Comparison of Computation Time of Best-Performing Models in All Three Frameworks

Framework	Model	Computation Time(min)
CSFL	DenseNet169	59.517
PLFL	MobileNet v2	18.34
FSSFL	MobileNet v2	31.54

of FSSFL provides 100% correct output for the healthy class, 59.38% correct output for the moderate class, and 20% correct output for the Seaver class. Finally, MobileNetV2 in PLFL provides 93.07% correct output for the healthy class, 72.73% correct output for the moderate class, and 100% correct output for the Seaver class. So we can see that the pseudo-labeling-based Federated Learning Framework was able to outperform the other two frameworks in moderate and severe classes.



TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	259 63.95%	11 2.72%	1 0.25%	271 95.57% 4.43%
Moderate	39 9.63%	55 13.58%	10 2.47%	104 52.88% 47.12%
Severe	0 0.00%	12 2.96%	18 4.44%	30 60.00% 40.00%
SUM	298 86.91% 13.09%	78 70.51% 29.49%	29 62.07% 37.93%	332 / 405 81.98% 18.02%

(a) Confusion Matrix of DenseNet169 in Client-Server-Based Federated Learning

TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	94 69.63%	6 4.44%	1 0.74%	101 93.07% 6.93%
Moderate	6 4.44%	16 11.85%	3 2.22%	25 64.00% 36.00%
Severe	0 0.00%	0 0.00%	9 6.67%	9 100.00% 0.00%
SUM	100 94.00% 6.00%	22 72.73% 27.27%	13 69.23% 30.77%	119 / 135 88.15% 11.85%

(b) Confusion Matrix of MobileNetV2 in Pseudo-Labeling-Based Federated Learning

TARGET OUTPUT	Healthy	Moderate	Severe	SUM
Healthy	98 72.59%	12 8.89%	0 0.00%	110 89.09% 10.91%
Moderate	0 0.00%	19 14.07%	4 2.96%	23 82.61% 17.39%
Severe	0 0.00%	1 0.74%	1 0.74%	2 50.00% 50.00%
SUM	98 100.00% 0.00%	32 59.38% 40.63%	5 20.00% 80.00%	118 / 135 87.41% 12.59%

(c) Confusion Matrix of MobileNetV2 in FixMatch-Based Federated Learning

Figure 5.12: Comparison of Performance in Different Federated Learning Approaches

So overall, in the three criteria stated here, pseudo-labeling-based Federated Learning outperforms the Client-Server-Based Federated Learning Framework and FixMatch-Based Federated Learning Framework with 0 label images in each client in the Knee Osteoarthritis Severity detection task.

Table 5.29: Comparison of Model Parameters Across Different Research Studies

Model Name	Number of parameter
EfficientNet-B1	7.8M
ResNet152V2	60.4M
InceptionResNetV2	28M
MobileNetV2 in Our method	3.4M

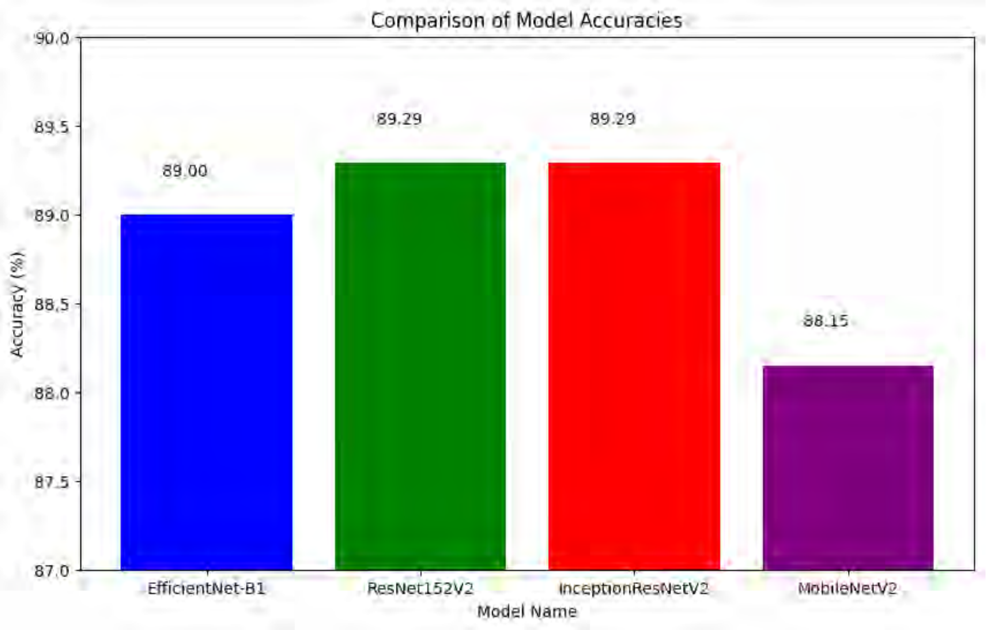


Figure 5.13: Comparison of Model Accuracies Across Different Architectures

When comparing our method (PLFL) to previous research in Table 5.29 and Figure 5.13, it's clear that our approach achieved performance that is very close to other models. The highest accuracy of 89.29% was achieved by InceptionResNetV2 and ResNet152V2 in the research conducted by H. Masood et. al[42], with these models having a large number of parameters 28 million and 60.4 million, respectively. Another study by B. C. Dharmani et al. [50] reported an accuracy of 89% with the EfficientNet-B1 model consisting of 7.8 million parameters. In contrast, our method using MobileNetV2, which has only 3.4 million parameters, achieved an accuracy of 88.15%. It's important to note that these other studies utilized fully supervised methods. Meanwhile, our approach employed semi-supervised techniques within a collaborative and federated setting. Despite having limited labeled data—930 healthy images, 329 moderate KAO images, and 93 severe images—our method achieved an accuracy that is only 1% less than the best-performing models mentioned. This demonstrates the efficiency and effectiveness of our approach, particularly given the constraints on labeled data.

Chapter 6

Conclusion

The primary objective of our research is to identify the most effective method for categorizing the severity of knee osteoarthritis with little human intervention. We have also emphasized the challenge of finding labeled medical photos due to the requirement for skilled individuals with expertise in this area. Additionally, this process is time-consuming and further compounded by a scarcity of medical professionals capable of doing this task. Due to these factors, the process of labeling medical photographs is expensive for an organization. Additionally, it is important to note that there are stringent regulations governing the sharing of medical pictures and data. Consequently, hospitals are reluctant to openly disclose medical data. Given these facts, we have implemented a novel solution known as the pseudo-labeling-based Federated Learning Framework. This approach effectively addresses the problem of annotation costs by utilizing zero-label data on clients. In addition, we have implemented Federated Learning, a method that addresses the problem of data sharing while maintaining the confidentiality of patients' information. Our technique is capable of resolving the adversarial assault known as data poisoning and incorrect annotation by the annotator, which can occur on the client side. By enhancing the train data with a high level of prediction confidence, we can exclude unreliable data, therefore serving the aim of data preparation.

The study compared a pseudo-labeling-based federated learning framework with two other frameworks: the traditional client-server-based federated learning framework and the FixMatch-based federated learning framework. The pseudo-labeling-based framework demonstrated superior model performance and time efficiency compared to both. The traditional federated learning framework included labeled data in all clients, while the FixMatch-based framework had 20% labeled data in each client. The Pseudo-labeling-based framework demonstrated exceptional performance with accuracy, weighted average precision, weighted average recall, and weighted average F1-score of 0.88. The MobileNetV2 model in the pseudo-labeling-based framework had the shortest duration for label generation and model training, 3.2 times shorter than the DenseNet169 model trained using the traditional Federated Learning Framework. The best-performing model from each framework was selected and compared. Furthermore, our system has exceptional prediction precision, especially in detecting moderate and severe instances of osteoarthritis, surpassing rival models. This is seen by the notable progress in accurately forecasting moderate and severe categories, highlighting the effectiveness of our method. This is the first instance where a pseudo-labeling-based federated learning system has been used to assess knee osteoarthritis severity from X-ray pictures. Through the utilization of pseudo-labeling in the context of Federated Learning, we are able to get unparalleled precision and effectiveness in assessing the severity of knee osteoarthritis based on X-ray images.

6.1 Limitations and Future Work

The primary obstacle is in acquiring data, especially within the healthcare sector where data is frequently sparse and restricted. Our pursuit brought attention to this fact, as we discovered the necessity for a larger dataset to carry out a comprehensive investigation. During our investigation, we observed a significant presence of data imprinting in each client, with a limited number of severe images in comparison to those classified as healthy or moderate.

In order to address this difficulty, we carefully allocated data points across clients, guaranteeing an equal quantity of photos on each server and maintaining overall balance. Nevertheless, as a result of the limits imposed by the hardware, we were restricted to a maximum of two customers, both of whom were equipped with identical setups.

In the future, we want to increase the number of clients and photographs by seeking the assistance of medical specialists to provide detailed explanations for knee osteoarthritis images. In addition, our strategy involves expanding our clients by providing them with non-iid datasets obtained from other geographical areas.

Alongside these efforts, our primary focus is on improving the weight aggregation approach to maximize the effectiveness of our study results.

Bibliography

[1] L. Solomon, P. Beighton, and J. S. Lawrence, "Rheumatic disorders in the south african negro. part ii. osteo-arthrosis," *South African medical journal = Suid-Afrikaanse tydskrif vir geneeskunde*, vol. 49, no. 42, pp. 1737–1740, 1975.

[2] M. A. Davis, W. H. Ettinger, J. M. Neuhaus, and W. W. Hauck, "Sex differences in osteoarthritis of the knee. the role of obesity," *American journal of epidemiology*, vol. 127, no. 5, pp. 1019–1030, 1988. DOI: 10.1093/oxfordjournals.aje.a114878.

[3] P. Dieppe, "Epidemiology of the Rheumatic Diseases Second Edition. AJ Silman, MC Hochberg (eds). Oxford: Oxford University Press, 2001, pp. 377, £95.00. ISBN: 0192631497.," *International Journal of Epidemiology*, vol. 31, no. 5, pp. 1079–1080, Oct. 2002, ISSN: 0300-5771. DOI: 10.1093/ije/31.5.1079-a. eprint: https://academic.oup.com/ije/article-pdf/31/5/1079/11434338/0311079a.pdf. [Online]. Available: https://doi.org/10.1093/ije/31.5.1079-a.

[4] D. Schiphof, M. Boers, and S. M. Bierma-Zeinstra, "Differences in descriptions of kellgren and lawrence grades of knee osteoarthritis," *Ann Rheum Dis*, vol. 67, no. 7, pp. 1034–1036, Jul. 2008, Epub 2008 Jan 15. PMID: 18198197. DOI: 10.1136/ard.2007.079020. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/18198197/.

[5] Institute of Medicine (US) Committee on Health Research and the Privacy of Health Information, "Hipaa, the privacy rule, and its application to health research," in *Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research*, S. Nass, L. Levit, and L. Gostin, Eds., Washington (DC): National Academies Press (US), 2009, ch. 4. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK9573/.

[6] Y. Zhang and J. M. Jordan, "Epidemiology of osteoarthritis," *Clin Geriatr Med*, vol. 26, no. 3, pp. 355–369, Aug. 2010, Erratum in: Clin Geriatr Med. 2013 May;29(2):ix. PMID: 20699159; PMCID: PMC2920533. DOI: 10.1016/j.cger.2010.03.001.

[7] D.-H. Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:18507866.

[8] N. A. Segal, M. C. Nevitt, K. D. Gross, *et al.*, "The multicenter osteoarthritis study: Opportunities for rehabilitation research," *PM & R: The Journal of Injury, Function, and Rehabilitation*, vol. 5, no. 8, pp. 647–654, 2013. DOI: 10.1016/j.pmrj.2013.04.014.

[9] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 `[cs.CV]`.

[10] G. Huang, Z. Liu, and K. Q. Weinberger, “Densely connected convolutional networks,” *CoRR*, vol. abs/1608.06993, 2016. arXiv: 1608.06993. [Online]. Available: http://arxiv.org/abs/1608.06993.

[11] A. G. Howard, M. Zhu, B. Chen, *et al.*, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *CoRR*, vol. abs/1704.04861, 2017. arXiv: 1704.04861. [Online]. Available: http://arxiv.org/abs/1704.04861.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, “Communication-Efficient Learning of Deep Networks from Decentralized Data,” in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, A. Singh and J. Zhu, Eds., ser. Proceedings of Machine Learning Research, vol. 54, PMLR, 2017, pp. 1273–1282. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html.

[13] P. Chen, *Knee osteoarthritis severity grading dataset*, Mendeley Data, V1, 2018. DOI: 10.17632/56rmx5bjcr.1.

[14] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L. Chen, “Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation,” *CoRR*, vol. abs/1801.04381, 2018. arXiv: 1801.04381. [Online]. Available: http://arxiv.org/abs/1801.04381.

[15] A. Tiulpin, J. Thevenot, E. Rahtu, J. Lehenkari, and S. Saarakkala, “Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach,” *Scientific Reports*, vol. 8, p. 1727, 2018. DOI: 10.1038/s41598-018-20132-7.

[16] D. Berthelot, N. Carlini, E. D. Cubuk, *et al.*, “Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring,” *CoRR*, vol. abs/1911.09785, 2019. arXiv: 1911.09785. [Online]. Available: http://arxiv.org/abs/1911.09785.

[17] A. Brahim, R. Jennane, R. Riad, *et al.*, “A decision support tool for early detection of knee osteoarthritis using x-ray imaging and machine learning: Data from the osteoarthritis initiative,” *Computerized Medical Imaging and Graphics*, vol. 73, pp. 11–18, 2019. DOI: 10.1016/j.compmedimag.2019.01.007.

[18] P. Chen, L. Gao, X. Shi, K. Allen, and L. Yang, “Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss,” *Comput Med Imaging Graph*, vol. 75, pp. 84–92, Jul. 2019, Epub 2019 Jun 13. PMID: 31238184; PMCID: PMC9531250. DOI: 10.1016/j.compmedimag.2019.06.002. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/31238184/.

[19] W. Li, F. Milletari, D. Xu, *et al.*, “Privacy-preserving federated brain tumour segmentation,” *CoRR*, vol. abs/1910.00962, 2019. arXiv: 1910.00962. [Online]. Available: http://arxiv.org/abs/1910.00962.

[20] Q. Xie, Z. Dai, E. H. Hovy, M. Luong, and Q. V. Le, “Unsupervised data augmentation,” *CoRR*, vol. abs/1904.12848, 2019. arXiv: 1904.12848. [Online]. Available: http://arxiv.org/abs/1904.12848.

[21] L. Biewald, *Experiment tracking with weights and biases*, Software available from wandb.com, 2020. [Online]. Available: https://www.wandb.com/.

[22] P. Cascante-Bonilla, F. Tan, Y. Qi, and V. Ordonez, "Curriculum labeling: Self-paced pseudo-labeling for semi-supervised learning," *CoRR*, vol. abs/2001.06001, 2020. arXiv: 2001.06001. [Online]. Available: https://arxiv.org/abs/2001.06001.

[23] X. Li, Y. Gu, N. Dvornek, L. H. Staib, P. Ventola, and J. S. Duncan, "Multi-site fmri analysis using privacy-preserving federated learning and domain adaptation: Abide results," *Medical Image Analysis*, vol. 65, p. 101 765, 2020, ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2020.101765. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841520301298.

[24] B. Liu, J. Luo, and H. Huang, "Toward automatic quantification of knee osteoarthritis severity using improved faster r-cnn," *International Journal of Computer Assisted Radiology and Surgery*, vol. 15, no. 3, pp. 457–466, 2020. DOI: 10.1007/s11548-019-02096-9.

[25] J. V. Pulido, S. Guleria, L. Ehsan, *et al.*, "Semi-supervised classification of noisy, gigapixel histology images," in *2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2020, pp. 563–568. DOI: 10.1109/BIBE50027.2020.00097.

[26] K. Sohn, D. Berthelot, C. Li, *et al.*, "Fixmatch: Simplifying semi-supervised learning with consistency and confidence," *CoRR*, vol. abs/2001.07685, 2020. arXiv: 2001.07685. [Online]. Available: https://arxiv.org/abs/2001.07685.

[27] L. Song, Y. Xu, L. Zhang, B. Du, Q. Zhang, and X. Wang, "Learning from synthetic images via active pseudo-labeling," *IEEE Transactions on Image Processing*, vol. 29, pp. 6452–6465, 2020. DOI: 10.1109/TIP.2020.2989100.

[28] M. Tan and Q. V. Le, *Efficientnet: Rethinking model scaling for convolutional neural networks*, 2020. arXiv: 1905.11946 [cs.LG].

[29] M. Abdul Salam, S. Taha, and M. Ramadan, "Covid-19 detection using federated machine learning," *PloS one*, vol. 16, no. 6, e0252573, 2021. DOI: 10.1371/journal.pone.0252573.

[30] N. Bayramoglu, M. T. Nieminen, and S. Saarakkala, "Automated detection of patellofemoral osteoarthritis from knee lateral view radiographs using deep learning: Data from the multicenter osteoarthritis study (most)," *Osteoarthritis and Cartilage*, vol. 29, no. 10, pp. 1432–1447, 2021. DOI: 10.1016/j.joca.2021.06.011.

[31] Q. Ding, Z. Li, and L. Zhao, "Feo content classification of sinter based on semi-supervised deep learning," in *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2021, pp. 640–644. DOI: 10.1109/CCDC52312.2021.9602424.

[32] Q. Dou, T. Y. So, M. Jiang, and et al., "Federated deep learning for detecting covid-19 lung abnormalities in ct: A privacy-preserving multinational validation study," *npj Digital Medicine*, vol. 4, p. 60, 2021. DOI: 10.1038/s41746-021-00431-6.

[33] X. Huang, X. Wang, W. Lv, *et al.*, *Pp-yolov2: A practical object detector*, 2021. arXiv: 2104.10419 [cs.CV].

[34] P. Kairouz, H. B. McMahan, B. Avent, *et al.* 2021.

[35] R. Mahum, S. U. Rehman, T. Meraj, *et al.*, “A novel hybrid approach based on deep cnn features to detect knee osteoarthritis,” *Sensors*, vol. 21, no. 18, p. 6189, 2021.

[36] A. Qayyum, K. Ahmad, M. A. Ahsan, A. I. Al-Fuqaha, and J. Qadir, “Collaborative federated learning for healthcare: Multi-modal COVID-19 diagnosis at the edge,” *CoRR*, vol. abs/2101.07511, 2021. arXiv: 2101.07511. [Online]. Available: https://arxiv.org/abs/2101.07511.

[37] P. Shi, J. Xin, and N. Zheng, “Correcting pseudo labels with label distribution for unsupervised domain adaptive vulnerable plaque detection,” in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 2021, pp. 3225–3228. DOI: 10.1109/EMBC46164.2021.9629833.

[38] T. Sun, D. Li, and B. Wang, “Decentralized federated averaging,” *CoRR*, vol. abs/2104.11375, 2021. arXiv: 2104.11375. [Online]. Available: https://arxiv.org/abs/2104.11375.

[39] Z. Yan, J. Wicaksana, Z. Wang, X. Yang, and K.-T. Cheng, “Variation-aware federated learning with multi-source decentralized medical image data,” *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2615–2628, 2021. DOI: 10.1109/JBHI.2020.3040015.

[40] L. Zhang, B. Shen, A. Barnawi, and et al., “Feddpgan: Federated differentially private generative adversarial networks framework for the detection of covid-19 pneumonia,” *Information Systems Frontiers*, vol. 23, pp. 1403–1415, 2021. DOI: 10.1007/s10796-021-10144-6.

[41] G. Im, “The concept of early osteoarthritis and its significance in regenerative medicine,” *Tissue Eng Regen Med*, vol. 19, no. 3, pp. 431–436, Jun. 2022, Epub 2022 Mar 4. PMID: 35244885; PMCID: PMC9130436. DOI: 10.1007/s13770-022-00436-6. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35244885/.

[42] H. Masood, E. Hassan, A. A. Salam, and M. Liaquat, “Osteo-doc: Kl-grading of osteoarthritis using deep-learning,” in *2022 2nd International Conference on Digital Futures and Transformative Technologies (ICoDT2)*, 2022, pp. 1–6. DOI: 10.1109/ICoDT255437.2022.9787470.

[43] P. Nguyen Huu, D. Nguyen Thanh, T. le Thi Hai, H. Chu Duc, H. Pham Viet, and C. Nguyen Trong, “Detection and classification knee osteoarthritis algorithm using yolov3 and vgg-16 models,” in *2022 7th National Scientific Conference on Applying New Technology in Green Buildings (ATiGB)*, 2022, pp. 31–36. DOI: 10.1109/ATiGB56486.2022.9984096.

[44] F. Prezja, J. Paloneva, I. Pölönen, and et al., “Deepfake knee osteoarthritis x-rays from generative adversarial neural networks deceive medical experts and offer augmentation potential to automatic classification,” *Sci Rep*, vol. 12, no. 1, p. 18 573, 2022. DOI: 10.1038/s41598-022-23081-4. [Online]. Available: https://doi.org/10.1038/s41598-022-23081-4.

[45] Z. Wang, A. Chetouani, D. Hans, E. Lespessailles, and R. Jennane, “Siamese-gap network for early detection of knee osteoarthritis,” in *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 2022, pp. 1–4. DOI: 10.1109/ISBI52829.2022.9761626.

[46] D. A. Zebari, S. S. Sadiq, and D. M. Sulaiman, “Knee osteoarthritis detection using deep feature based on convolutional neural network,” in *2022 International Conference on Computer Science and Software Engineering (CSASE)*, 2022, pp. 259–264. DOI: 10.1109/CSASE51777.2022.9759799.

[47] M. Zenk, D. Zimmerer, F. Isensee, P. F. Jäger, J. Wasserthal, and K. Maier-Hein, “Realistic evaluation of fixmatch on imbalanced medical image classification tasks,” in *Bildverarbeitung für die Medizin 2022*, K. Maier-Hein, T. M. Deserno, H. Handels, A. Maier, C. Palm, and T. Tolxdorff, Eds., Wiesbaden: Springer Fachmedien Wiesbaden, 2022, pp. 291–296, ISBN: 978-3-658-36932-3.

[48] D. Cai, S. Wang, Y. Wu, F. X. Lin, and M. Xu, “Federated few-shot learning for mobile nlp,” in *Proceedings of the 29th Annual International Conference on Mobile Computing and Networking*, ser. ACM MobiCom ’23, ACM, Oct. 2023. DOI: 10.1145/3570361.3613277. [Online]. Available: http://dx.doi.org/10.1145/3570361.3613277.

[49] P. Dhade and P. Shirke, “Federated learning for healthcare: A comprehensive review,” *Engineering Proceedings*, vol. 59, no. 1, 2023, ISSN: 2673-4591. DOI: 10.3390/engproc2023059230. [Online]. Available: https://www.mdpi.com/2673-4591/59/1/230.

[50] B. C. Dharmani and K. Khatri, “Deep learning for knee osteoarthritis severity stage detection using x-ray images,” in *2023 15th International Conference on COMmunication Systems & NETworkS (COMSNETS)*, 2023, pp. 78–83. DOI: 10.1109/COMSNETS56262.2023.10041355.

[51] O. Kurasova, V. Medvedev, A. Šubonienė, *et al.*, “Semi-supervised learning with pseudo-labeling for pancreatic cancer detection on ct scans,” in *2023 18th Iberian Conference on Information Systems and Technologies (CISTI)*, 2023, pp. 1–6. DOI: 10.23919/CISTI58278.2023.10211356.

[52] A. S. Mohammed, A. A. Hasanaath, G. Latif, and A. Bashar, “Knee osteoarthritis detection and severity classification using residual neural networks on preprocessed x-ray images,” *Diagnostics*, vol. 13, no. 8, p. 1380, 2023. DOI: 10.3390/diagnostics13081380. [Online]. Available: https://doi.org/10.3390/diagnostics13081380.

[53] L. Qiu, J. Cheng, H. Gao, W. Xiong, and H. Ren, “Federated semi-supervised learning for medical image segmentation via pseudo-label denoising,” *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 10, pp. 4672–4683, 2023. DOI: 10.1109/JBHI.2023.3274498.

[54] T. Tariq, Z. Suhail, and Z. Nawaz, “Knee osteoarthritis detection and classification using x-rays,” *IEEE Access*, vol. 11, pp. 48 292–48 303, 2023. DOI: 10.1109/ACCESS.2023.3276810.

[55] L. Yang, L. Qi, L. Feng, W. Zhang, and Y. Shi, “Revisiting weak-to-strong consistency in semi-supervised semantic segmentation,” 2023. arXiv: 2208.09910 [cs.CV].

[56] S. Zhou, S. Tian, L. Yu, *et al.*, "Fixmatch-ls: Semi-supervised skin lesion classification with label smoothing," *Biomedical Signal Processing and Control*, vol. 84, p. 104 709, 2023, ISSN: 1746-8094. DOI: 10.1016/j.bspc.2023.104709. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809423001428.

[57] Y. Chen, C. Zhang, Y. Ke, *et al.*, "Semi-supervised medical image segmentation method based on cross-pseudo labeling leveraging strong and weak data augmentation strategies," 2024. arXiv: 2402.11273 `[cs.CV]`.

[58] H. Hsu and R. M. Siwiec, "Knee osteoarthritis," in *StatPearls*, Updated 2023 Jun 26. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2024 Jan-., Jun. 2024. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK507884/.

[59] Keras, *Keras: The python deep learning api*, https://keras.io/, Accessed: 2024-05-18, 2024.

[60] *OpenCV - open source computer vision library*, https://opencv.org/, Accessed: 2024-05-18.

[61] *Tensorflow*, https://www.tensorflow.org/, Accessed: May 18, 2024.

[62] *Tf.keras.applications.densenet169*, https://www.tensorflow.org/api_docs/python/tf/keras/applications/DenseNet169, Accessed: May 18, 2024.

[63] *Tf.keras.applications.densenet201*, https://www.tensorflow.org/api_docs/python/tf/keras/applications/DenseNet201, Accessed: May 18, 2024.

[64] *Tf.keras.applications.mobilenetv2*, https://www.tensorflow.org/api_docs/python/tf/keras/applications/MobileNetV2, Accessed: May 18, 2024.