Identification of Charcot's Feet (Diabetic Foot) using AI & Medical Images

By

FAWAZ NADEEM

Fall-2020-MSCSE 00000328009

Supervised by

Dr. Zartasha Mustansar

Department of Engineering

School of Interdisciplinary Engineering & Sciences (SINES)

National University of Sciences & Technology (NUST)

June 2023

Annex A to NUST Letter No.

0972/102/Exams/Thesis-Cert

dated 23 Dec 16.

THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS/MPhil thesis written by Mr **_FAWAZ NADEEM_** Registration No. **_00000328009_** of **_SINES_** has been vetted by undersigned, found complete in all aspects as per NUST Statutes/Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS/MPhil degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature with stamp: _____

Name of Supervisor: _Dr. Zartasha Mustansar_

Date: _____

Signature of HoD with stamp: _____

Date: _____

Countersign by

Signature (Dean/Principal): _____

Date: _____

Dedication

"To my beloved family, who has supported me every step of the way, thank you for your unwavering love and encouragement. To my teacher, who has guided me through this academic journey, thank you for sharing your knowledge and inspiring me to strive for excellence. To my friends, who have cheered me on and kept me sane during stressful times, thank you for being a constant source of joy and laughter in my life. This thesis is dedicated to all of you, with the hope that my achievements will be a source of pride and happiness for us all."

ACKNOWLEDGMENT

"I would like to take this opportunity to express my heartfelt gratitude to everyone who has helped me in completing this thesis. First and foremost, I would like to acknowledge myself for the hard work, dedication, and perseverance that I have put into this project.

I would also like to express my sincere appreciation to my supervisor, Dr. Zartasha Mustansar, for her invaluable guidance, support, and motivation throughout this research. Her expertise, constructive criticism, and encouragement have been instrumental in shaping my ideas and enhancing the quality of my work.

I would like to extend my thanks to the members of my Graduate Examination Committee (GEC) for their valuable feedback and suggestions that have helped me in improving the quality of this thesis. I am also grateful to the Head of the Department (HOD) Dr. Mian Ilyas and the Principal Dr. Hammad Mahmood Cheema for providing me with the necessary resources and facilities that were essential for the successful completion of this research.

Finally, I would like to express my appreciation to the Department of SINES for providing me with an excellent academic environment and giving me the opportunity to pursue my research in a supportive and stimulating community. This thesis would not have been possible without the collective efforts of all those who have supported and encouraged me along the way. Thank you."

Declaration

I, *Fawaz Nadeem* declare that this thesis titles “Identification of Charcot’s Feet using AI and Medical Images” and the work presented in it are my own and has been generated by me as a result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a Master of Science degree at NUST.
2. Where any part of this Thesis has previously been submitted for a degree or any other qualification at NUST or any other institution, this has been clearly stated.
3. Where I have consulted the published work of others, this is always clearly attributed.
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
5. I have acknowledged all main sources of help.
6. Where the thesis is based on work on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Fawaz Nadeem,

00000328009

Copyright Notice

Table of Contents

List of Tables

List of Figures

ABSTRACT

Diabetes, one of the world's most common diseases, poses a significant threat to overall health. Among its major health concerns is the depletion and weakening of bones, leading to conditions like Charcot Foot or Diabetic Foot. Despite the global prevalence of diabetes and its impact on millions of individuals, our understanding of its effects on bone health remains limited. Addressing this knowledge gap becomes increasingly crucial which forms the basis of this study. Diabetic foot cases often suffer from deteriorating bone health, resulting in increased bone curvature, mechanical instability, and porosity.

This study seeks to investigate mechanics associated with diabetic foot, its analysis and employs Artificial intelligence (AI) effectively to address early prognosis of this disease. Generative Adversarial Networks (GANs) is used as a preferred methodology in this work to produce synthetic data since obtaining rich data in these cases is challenging specially from the hospitals in Pakistan. A total of 560 images were synthetically reproduced to assess reliability. The reliability of this data was then evaluated using Fréchet's Inception Distance (FID) and Inception Score (IS) to address validation. These Images were used to train multiple U-net models, which were further assessed and compared using the Intersection over Union (IOU) metric.

Our study findings demonstrate the potential of AI-generated synthetic images in U-net models for accurately identifying the advancement of bone damage in diabetic feet. Notably, the Recurrent Residual (R2) U-Net model outperforms other models by effectively detecting the progression of the disease on real x-ray images, achieving a significant average IOU score of 0.75 which is in coherence with the literature published. These results hold valuable implications for clinicians, as they can utilize our findings for early prognosis of this condition, facilitating timely intervention and management strategies.

Keywords: Charcot Foot (CF), Diabetic Foot (DF), GANs, AI, IOU, FID, IS

1 INTRODUCTION

Foot issues are a significant health concern for individuals with diabetes, resulting from long-term damage to the blood vessels and nerves in the feet due to uncontrolled blood sugar levels. Diabetic neuro-arthropathy manifests as foot tingling, numbness, discomfort, and lack of sensation, with CF representing a complex complication of neuropathy and diabetes. CF inflicts destructive effects on the foot and ankle, characterized by uncontrolled cycles of inflammation, ultimately resulting in a rocker-bottom deformation that can be prevented by early diagnosis and treatment. First identified by French physician and neurologist Jean Martin Charcot in 1868 [1], CF was associated with diabetes mellitus in 1936, posing a challenge for skilled practitioners [2] to diagnose. Diagnostic imaging techniques including X-rays, CT scans, nuclear medicine scintigraphy, positron emission tomography, and magnetic resonance imaging, play a crucial role in diagnosing CF, with these techniques providing distinct differences from osteomyelitis. A lot of techniques are available in literature for understanding how damage progress in CF. However, the management of diabetic feet is still a complex and challenging task for healthcare professionals. Artificial Intelligence (AI) faces similar challenges due to the multifactorial nature of the disease. Diabetic foot ulcers result from a combination of peripheral neuropathy, peripheral arterial disease, and infections, making it more difficult for AI to develop a comprehensive understanding of the condition. Additionally, diabetic foot ulcers can vary widely in their presentation, making it hard for AI algorithms to recognize and classify them accurately. This is why this problem was undertaken as a case study in this thesis, to understand how AI can be effectively used for detection and classification of this specific problem.

AI systems require large amounts of high-quality data to train and develop models that can accurately identify and manage diabetic foot ulcers, which may be challenging to obtain due to the limited availability of well-curated datasets specially across Pakistan. Finally, AI may not be able to replace the experience and judgment of healthcare professionals who have years of experience in managing diabetic foot ulcers. However, it may serve as an assistive technique for better prognosis & diagnosis.

1.1 Background and motivation

CF, which has been linked to diabetes mellitus and affects the joints and bones (single/multiple involvement), was originally identified in 1883 [3]. Bone regression is one of fundamental health concepts being researched upon. Deterioration of bone health can be brought on by illness, cancer, etc. Just a very brief recognition is made of the idea that diabetes affects bone health in relatively underdeveloped (third world) nations. Abnormal blood sugar levels lead to multiple neuropathic problems. The foot and ankle's soft tissue, joints, and bones are all impacted by osteoarthritis neuropathies. It has been determined that diabetes does weaken bones, which has been linked to the "rock bottom" foot syndrome, which causes the midfoot to collapse. Joint and bone deformities result from this. Pain and discomfort characterize the acute period, which fades over time as progressive damage occurs. Age and how diabetes care is provided are more important factors in determining risk of developing CF than the type of diabetes [1]. Although polyneuropathy is the underlying cause of the illness, its precise pathogenic mechanism is still unknown. The condition primarily affects people between the ages of 45 and 60.

Based on this foundation, our thesis aims to explore the applicability of integrating our research with existing image-based visual techniques. By leveraging the

remarkable advancements in artificial intelligence (AI) and machine learning algorithms, which have revolutionized various fields, we can effectively design and implement vital healthcare solutions with optimized resource utilization. Furthermore, we will examine the existing body of research on the underdiagnosed CF disease and propose a methodology that incorporates relevant methods identified through an extensive literature review.

Currently, the prevalence statistics for a particular disease are relatively low, and timely detection plays a crucial role in preserving the affected foot of diabetic individuals. Advancements in AI and machine learning have opened up new possibilities for computational designs to support healthcare scenarios. In our study, the utilization of a tool for early detection of this disease would significantly contribute to reducing its spread. Computational imaging processes offer promising avenues for addressing identification challenges, and we aimed to develop an open-source solution to leverage the emerging field of AI to its fullest potential. Additionally, we seek to raise awareness about this under-diagnosed situation, emphasizing the importance of providing accessible and doctor-friendly diagnostic tools to enhance efficiency in critical healthcare decisions. This endeavor not only represents a step towards integrating computational assistance into medical practices but also broadens our understanding of the impact of diabetes on bone health.

1.2 Objective of study

The study Aims to create a model that will aid in identifying the initial stage of CF so that precautions can be taken to avoid progression of the disease.

1) Understanding & generating augmented data to address, reliability in synthetic images.

2) a) Detection and b) Analysis of bone damage progression of Charcot's feet using computing, AI, and medical imaging.

1.3 Problem statement/Research statement

We attempt to answer the question that can AI and ML using synthetic images identify bone damage caused by diabetes.

In our research, we are determined to experiment for conclusive findings. We aim to utilize the concepts of creating augmented relative images to increase the dataset in demand. We have the goal to identify the early prognosis of bone damage due to diabetes using AI assisted medical imaging. We shall utilize machine learning models to detect the required changes in our imaging identification case.

1.4 Importance to the society

The demand of society has increased to acquire quick solutions to evade health problems that are unaware to society and pose a threat to their health. By harnessing the potential of AI and machine learning algorithms, society can benefit from enhanced healthcare solutions. Integrating these technologies with image-based visual techniques can lead to more accurate and efficient diagnosis, enabling early detection and treatment of diseases such as CF. This in turn, can improve a patient's outcomes and quality of life.

The utilization of AI integrated computational techniques can help streamline healthcare processes and optimize resource allocation. By mankind informed judgements and leveraging computational tools, healthcare professionals can make effective use of limited resources, ensuring that essential healthcare solutions are delivered to society efficiently and costly-effectively.

The study specifically focuses on an underdiagnosed disease, CF. By conducting research and developing methodologies to improve its detection and diagnosis, the thesis contributes to addressing the challenges associated with underdiagnosed conditions, leading to increased awareness, earlier interventions, and improved healthcare decision-making for individuals affected by such diseases.

This will also contribute to the expansion of knowledge not only to academic body but also facilitates knowledge sharing and dissemination within the scientific community. By building upon previous research, society benefits from cumulative understanding of the subject matter and can make further advancements in the field.

1.5 Contribution of this study

1) This thesis presents a comprehensive exploration and illustration of a systematic approach, employing an open-source software pipeline, to develop an identification model for a disease that is frequently underdiagnosed.
2) It introduces an innovative amalgamation of four key components: Generative Adversarial Networks (GANs), metrics for assessing synthetic image quality, segmentation models, and evaluation metrics.
3) By utilizing the open-source pipeline, synthetic images are generated to serve as training datasets for constructing a prediction model using real-world datasets. Though not reliable, but good where the acquisition of data becomes really challenging.

1.6 Organization of thesis

This thesis is structured into distinct chapters.

Chapter 1 serves as the introduction, presenting an overview of the research.

Chapter 2 provides an in-depth analysis of the relevant literature necessary for comprehending this study.

Chapter 3 elucidates the methodology employed, outlining the chosen research approach and pipeline.

Chapter 4 scrutinizes the obtained results.

Chapter 5 concludes with a comprehensive discussion and outlines potential avenues for future research.

In conclusion, we posit that Generative Adversarial Networks (GANs) present a viable approach to augmenting our dataset for effective model training. To enhance our understanding, we systematically executed a series of experimentation and exploration, leveraging existing pathways already investigated within our research framework. Our endeavors primarily involved harnessing readily accessible open-source algorithms and models found in programming language libraries.

2 LITERATURE REVIEW

2.1 Overview of Charcot foot

In this study, due to long-term damage caused by uncontrolled blood sugar levels to the blood arteries and nerves in the feet, foot problems are a major worry for persons with diabetes. Diabetes-related neuroarthropathy can result in tingling, numbness, pain, and a lack of sensation in the feet. The complex condition of CF can cause devastating cycles of inflammation and alter the shape of the foot, giving it a rocker-bottom deformity. It is a consequence of neuropathy and diabetes. Even for experienced doctors, diagnosing CF can be difficult. However, imaging methods like X-rays, CT scans, nuclear medicine scintigraphy, positron emission tomography, and magnetic resonance imaging might be helpful. The normal course of CF is inflammation-fragmentation-coalescence-remodeling. Diabetes often results in foot ulcers brought on by neuropathies and ischemia, and CF-related bone regression can damage the forefoot, ankle, hindfoot, and Centre of the foot joints. If the stage of deformation has progressed past the point of no return, the only recommended treatment is to have the sick foot removed as soon as possible to prevent the disease from spreading.

2.1.1 Etiology

One of the most well-known medical, financial, and societal issues is diabetes. By estimates of International Diabetic Federation, one in four adults (26.7%) in Pakistan are living with diabetes, the highest national prevalence in the world [4]. According to the IDF, 6.7 million people in Pakistan have diabetes, and that number is expected to rise to 12.8 million by 2035 [3]. It reduces a patient’s quality of life and affects social participation [5]. Diabetes also comes with complications or associated risks such as

infection, foot ulceration, and tissue loss in the foot, as well as factors that contribute to morbidity and mortality [5]. The illness caused consists of infection, microvascular, and neuropathic symptoms. Diabetes is brought on by restricted insulin production, which raises blood sugar levels and poses serious risks to one's health or even life. Higher glucose levels and hyperglycemia trigger inflammatory reactions and compromise cellular immunity. Age, the severity of the illness, and poor glycemic control over several years are all risk factors [5]. Therefore, diabetic individuals put themselves at risk for developing limb-threatening foot infections and uncontrollable diabetes that affects soft tissue and uncontrollable hyperglycemia, which can result in osteitis[5]. There are two main forms of diabetes: type 1 diabetes, which results in complete insulin shortage and type 2 diabetes that is insulin resistance [6]. Certain factors, including genetics, obesity, physical inactivity, and certain medical conditions, can risk of developing diabetes [6].

2.1.2 Pathogenesis

Inflammation-Fragmentation-Coalescence-Remodeling is the disease's natural progression. To prevent foot deformity, the standard course of therapy is to plaster the foot with individualized detachable total contact casts. Reduced pressure distribution can be achieved by wearing orthopedic shoes and using their special insoles [5]. The only advised treatment is to have the diseased foot removed as soon as possible to stop the disease from spreading if the stage of deformation has advanced past the point of no return.

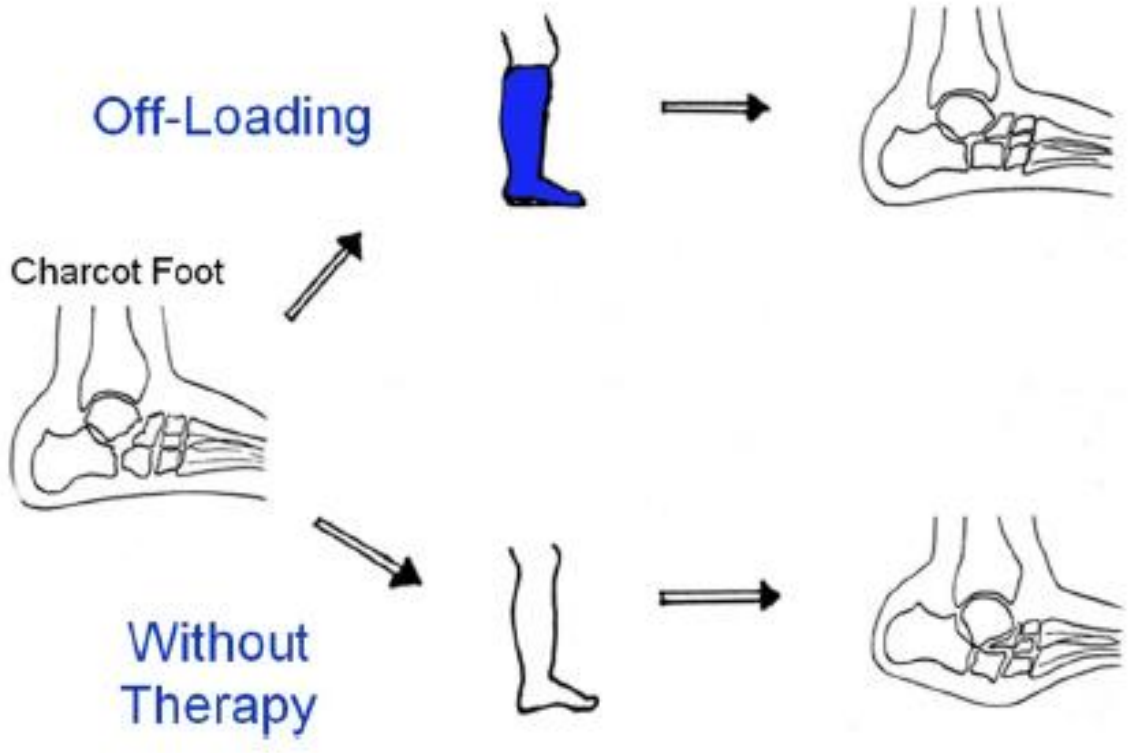


Figure 1 The patient has a better chance of recovering adequately with off-loading therapy and total contact casts if they have an intact longitudinal arch and avoid crippling abnormalities [1].

The condition is tracked and diagnosed using standard radiography. To evaluate the bones while under strain. The measurements on radiographs aid in determining the severity of the CF deformity.

2.1.3 Foot Ulcers

Amputation risks are increased because of diabetic foot ulcer complications. This illness thus tackles the related medical, financial, and societal effects.

Diabetic peripheral artery disease, neuropathy, or infection make up the majority of complexity. However, there is still more research to be done on bone regression brought on by its nature. It consists of both ischemia and neuropathy. Diabetes-related foot ulcers are caused by many neuropathies that impede the ability to feel pain and temperature.

2.1.4 Bone regression

Foot complications are said to be the most common cause of hospital admissions and lower limb subject [3]. Research focuses on the bone regression that occurs in CF. Joint deterioration happens over time. commonly impacts the forefoot, ankle, hindfoot, and Centre of the foot joints. 13% of people have diabetes on average [3]. The swelling of the foot brought on by inflammation results in osteolysis, which is indirectly to blame for the gradual fracture and dislocation that characterize the condition. Most evidence is circumstantial. It has been suggested that the development of bone and joint degeneration in neuropathic individuals is caused by a neurally driven response that increases peripheral blood flow and actively resorbs bone. Midfoot is primarily impacted by Charcot neuroarthropathy. The Sanders classification highlights the midfoot as the underlined area affected by Charcot disease [7]. When evaluating intraarticular calcaneal fractures involving the posterior facet of the calcaneus, the Sanders classification system is employed. It is based on the location of intraarticular fracture lines on semi-coronal CT scans. Not only used for fracture pattern identification but it’s also used to predict outcome [8].

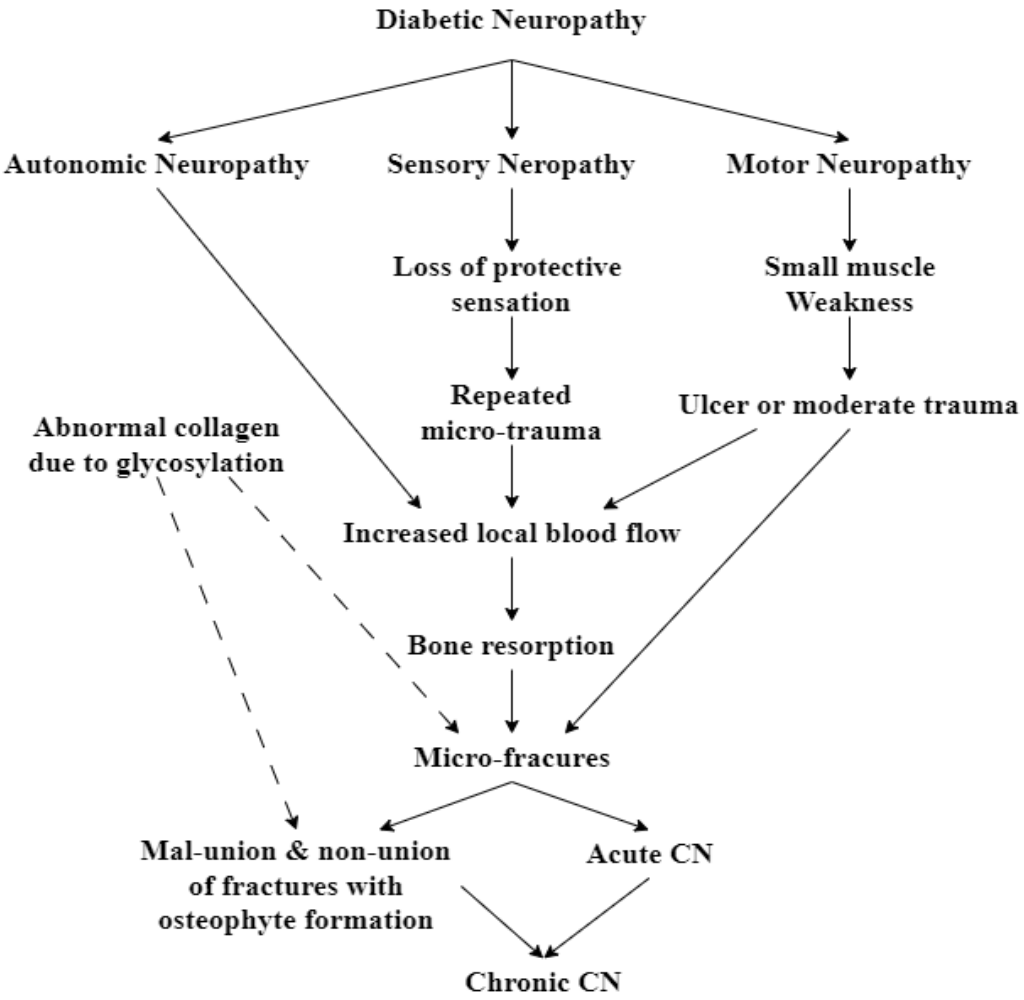


Figure 2 Diagram showing possible pathogenic processes that could lead to Charcot neuroarthropathy (CN). Increased blood flow and neuropathy both seem to be significant, and aberrant collagen structure may also be at play [9].

Infection and ulceration brought on by Charcot neuroarthropathy culminate in limb amputation. Thus, to stop the problem's rapid progression toward foot deformity, involvement is required to identify the condition early. Bone distortion is a result of continuous inflammation [6]. This results in bone reabsorption and fractures at multiple places.

It is classified into stages:

- **Stage 0** (Prodromal): Edematous foot with or without bounding pulses and temperature elevation. Not many bodily changes are seen.
- **Stage 1** (Developmental, acute): A brief period of trauma causes bone fragmentation, along with joint subluxation and dislocation. Immediate medical confirmation is required for an early diagnosis; failure to do so results in a permanent change in bone form.
- **Stage 2** (Subacute): Healing of fractures and decreased edema.

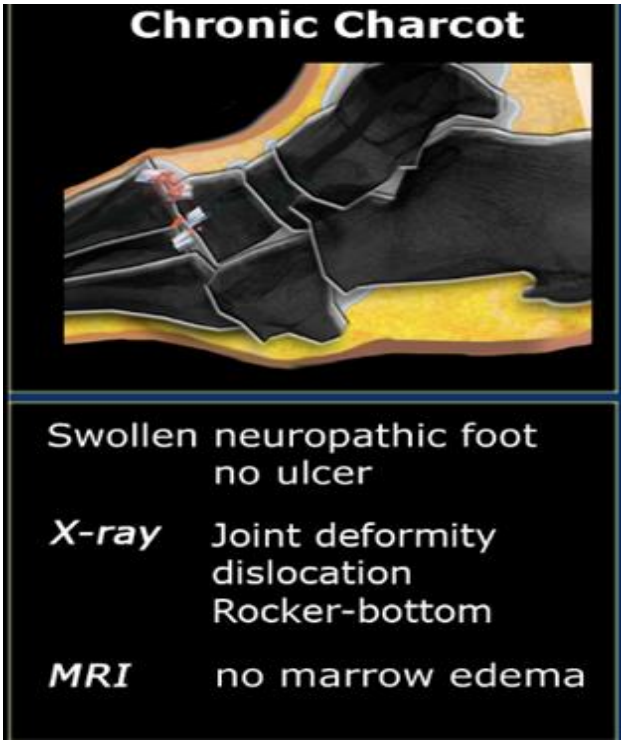


Figure 3 Chronic stage (Extremely Alarming).Description of diagnosis via X-ray and MRI imaging techniques[10].

- **Stage 3** (Chronic): Bone is remodeled. Three types of Charcot foot classified: anatomical, clinical, and radiological.

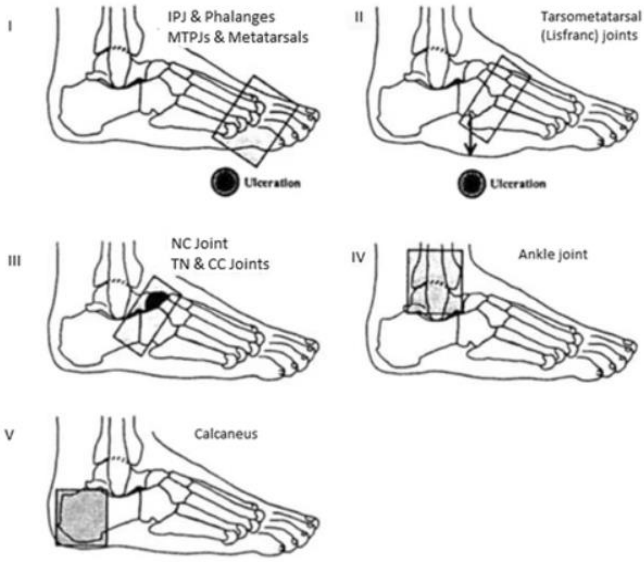


Figure 4 The CF is staged on the anatomical placement of the foot in Rogers and Fryberg's work. The first stage of five is displayed. Stage 1 causing IPJ & Phalamges, MTPJs, and skeletally damaging alterations; Stage 2 targets the Tarsometatarsal joints (ulceration) at the apex of collapsed cuboid cuneiforms; Stage 3 targets the naviculocuneiform, calcaneocuboid, and talonavicular joints with fragmentation of CC, TN, and NC joints; Stage 4 targets the ankle joint; and Stage 5 targets the calcaneal bone [11].

2.1.5 Charcot foot stages

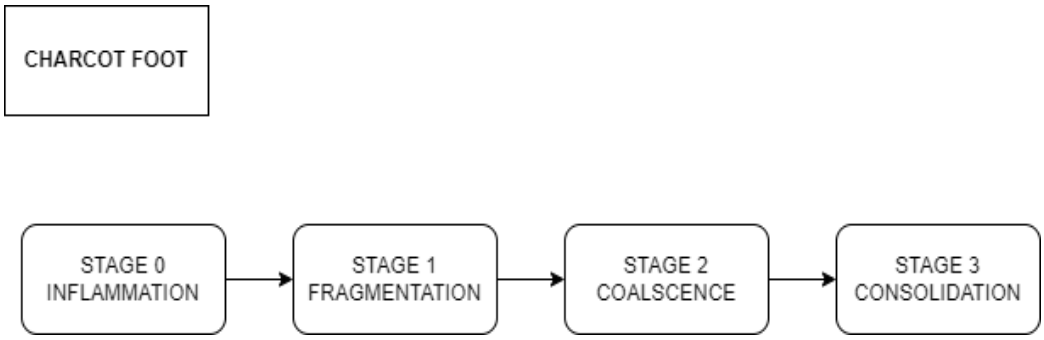


Figure 5 CF: natural course of disease.

- **Stage 0: Prodromal**

This happens when a patient with neuropathy exhibits foot alterations such as edema, redness, warmth, and pain that are equivalent to inflammation. These are the early warning indications of foot architecture collapse in CF later phases. When treating patients with diabetes mellitus, endocrinology engulfs the doctor, therefore essential knowledge of stage 0 symptoms is necessary [12]. The stage is recognized with fragmentation, dislocation, and osseous debris. Clinically, stage 0 consists of

edematous, erythematous warm foot. Where the foot temperature increases high with minimal effort such as walking for 50 steps or less with a significant increase of about 60% [13].

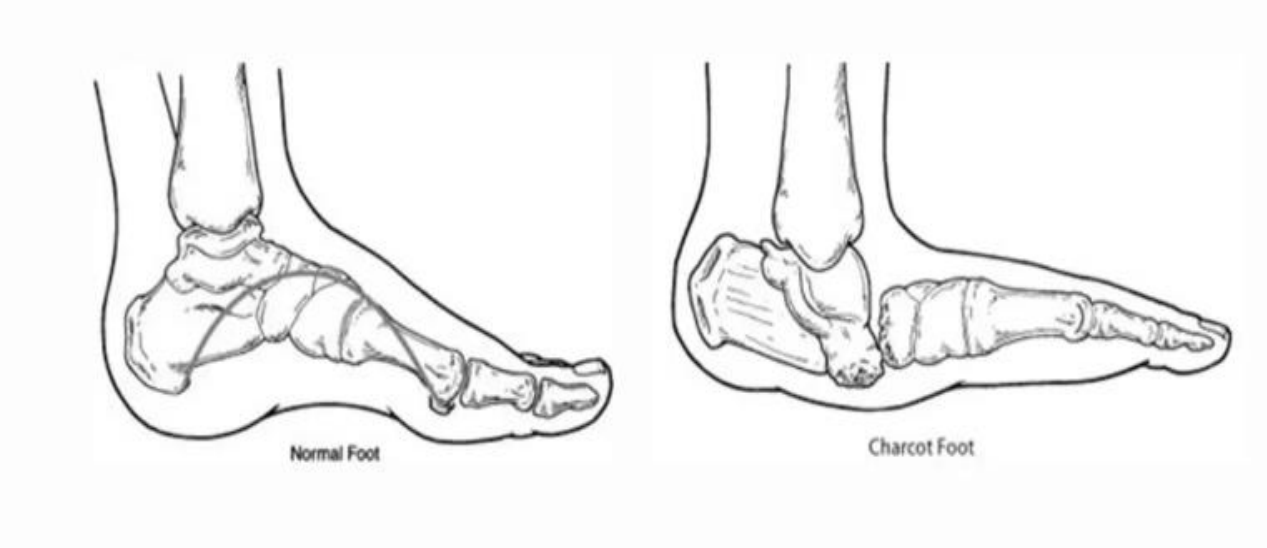


Figure 6 The malformation that might result in a CF is demonstrated in this image. This happens because the foot bones weaken and lose their ability to feel, which makes it possible for a fracture to go undetected by the person who is suffering from it. The foot will continue to distort without casting or splinting [14].

- **Stage 1: Development, Acute**

The Acute stage is characterized by erythema, increased foot temperatures, and clinically convincing edema that looks like grout. Here, bone fragmentation takes place, marked by periarticular fractures and displaced joints, leading to a foot deformation. This suggests bone fragmentation occurs during this stage, accompanied by periarticular fractures and displaced joints, which ultimately lead to foot deformation. This suggests a progressive nature of the condition, where structural damage occurs to the bones and joints, resulting in deformity.

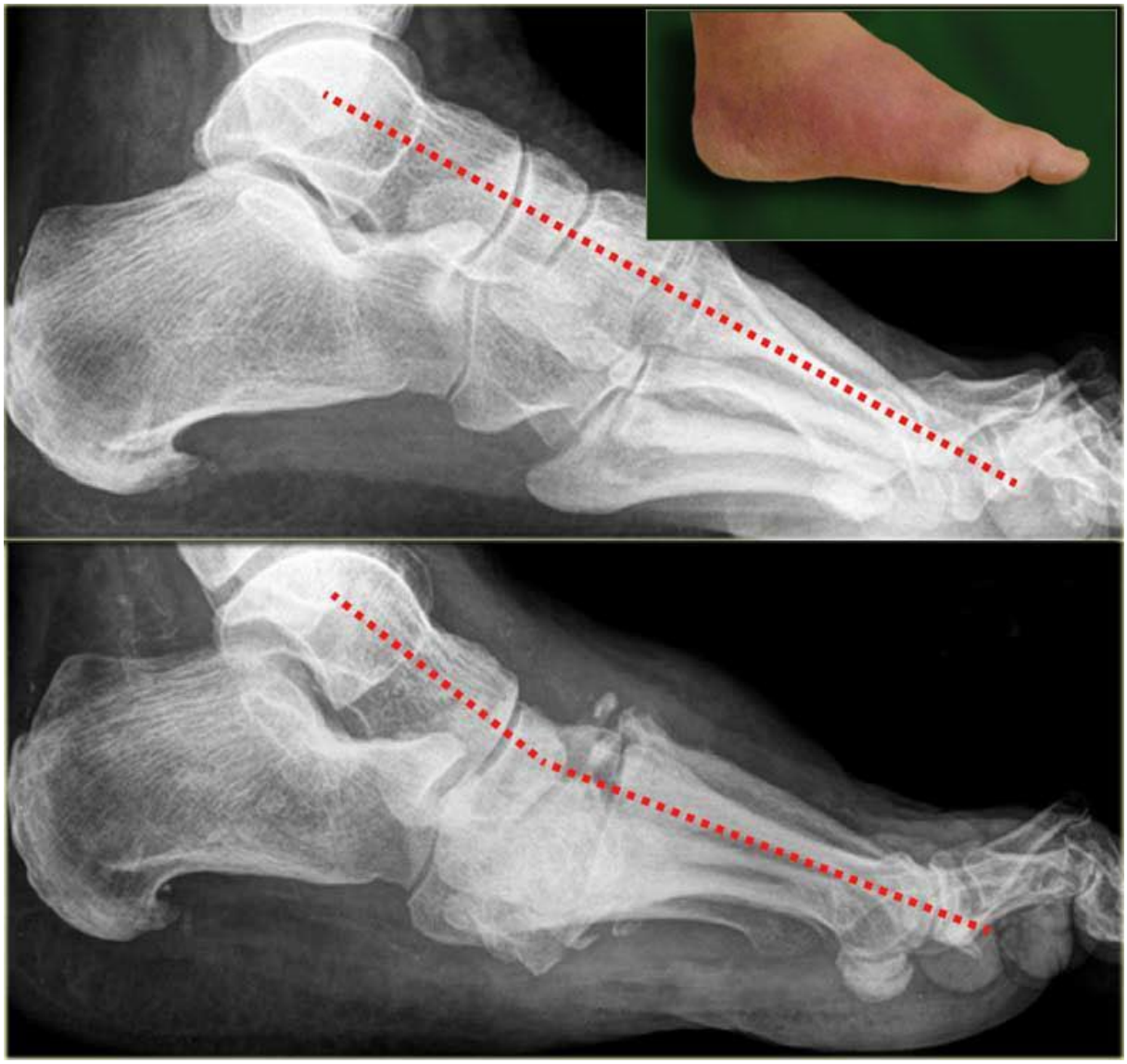


Figure 7 The fast and gradual degeneration of bones and joints during the acute stage of Charcot neuro-osteoarthropathy can be seen in a matter of days or weeks. Total contact casting can immobilize an individual and stop additional bone and joint damage. An image of a patient with diabetic neuropathy and a scorching foot is seen here. The radiographs are normal in the acute stage and may not rule out the diagnosis of acute Charcot neuro-osteoarthropathy.

- **Stage 2: Subacute**

The damaged, shattered bones consolidate (also known as coalesce) during this stage. Radiographs will indicate the bone's healing stage when the warmth, swelling, and redness start to fade. If the essential precautions are not followed at this stage, the foot structure will permanently distort.

- **Stage 3: Chronic**

In this last stage of the illness, healing takes place, leading to reconstruction and consolidation. The normal contour of the foot has changed from how it appeared initially, even if it is possible to say that the bones have healed because the temperature, redness, and swelling are no longer present. The act of stabilizing a foot by fusing bone pieces together is an example of consolidation, which results in a stable but malformed foot [15].

Figure 8 X-ray of a deformed (Charcot) foot [1].

The feet area in question for research is:

- Meary's angle9: a normal angle of roughly 0 degrees between the line passing through the longitudinal axis of the first metatarsal and the line coming from the middle of the body of the talus, bisecting the talar neck and head.
- Cuboid height: distance along a line drawn from the calcaneal tuberosity's plantar surface to the cuboid's plantar aspect.

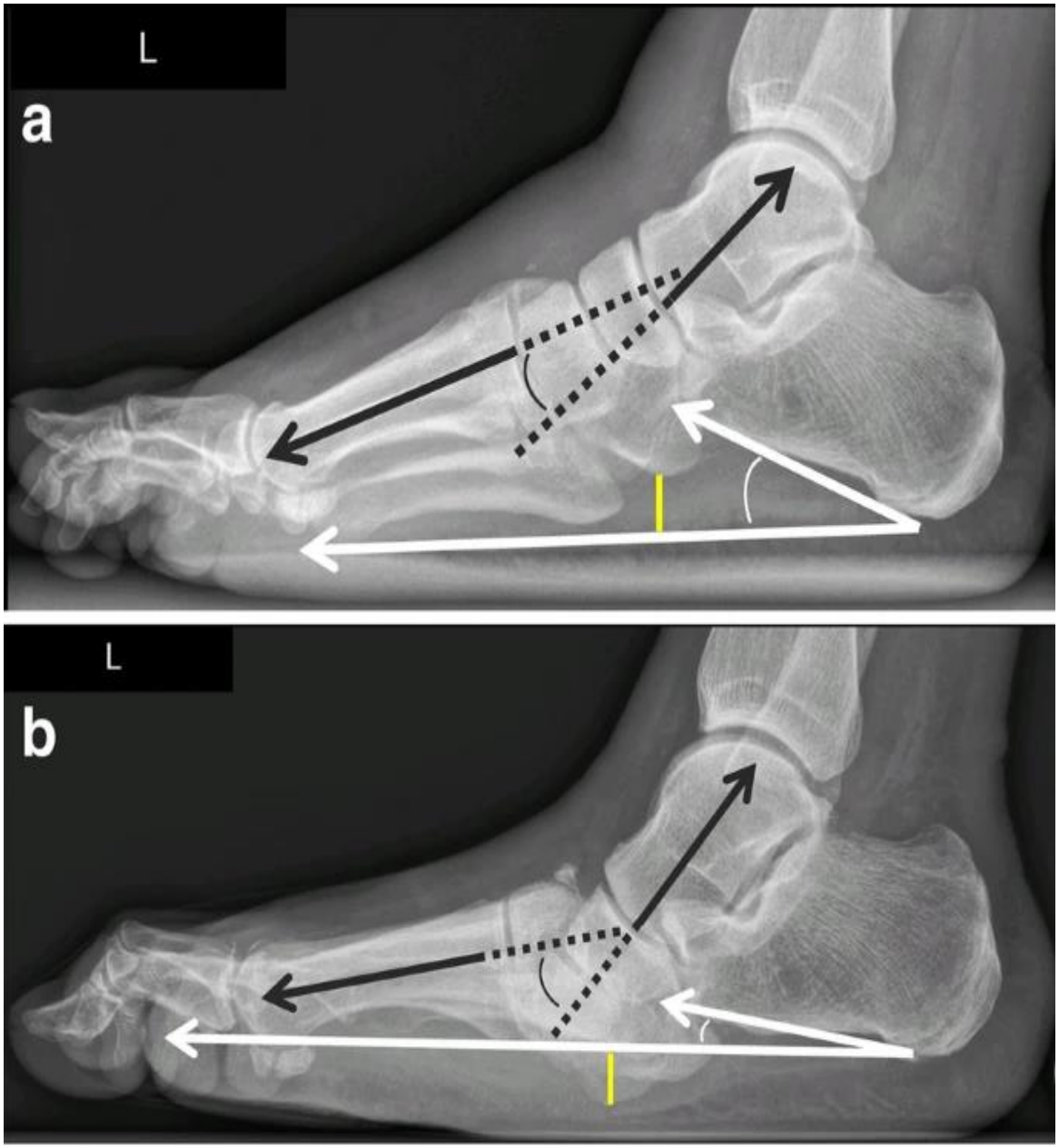


Figure 9 Lateral weight-bearing radiographs depict the progression of CF disease in a consistent pattern across time (a baseline, b 10 months later). Take note of Meary's angle (black angle), which is steadily increasing, the shrinkage of cuboid height, which is turning negative, and the reduction of calcaneal pitch (white angle) [1].

The disease's main focal starts from the midfoot area that spreads in time, affecting the whole foot and in turn keeps on spreading until further notice as shown in Figure 8.

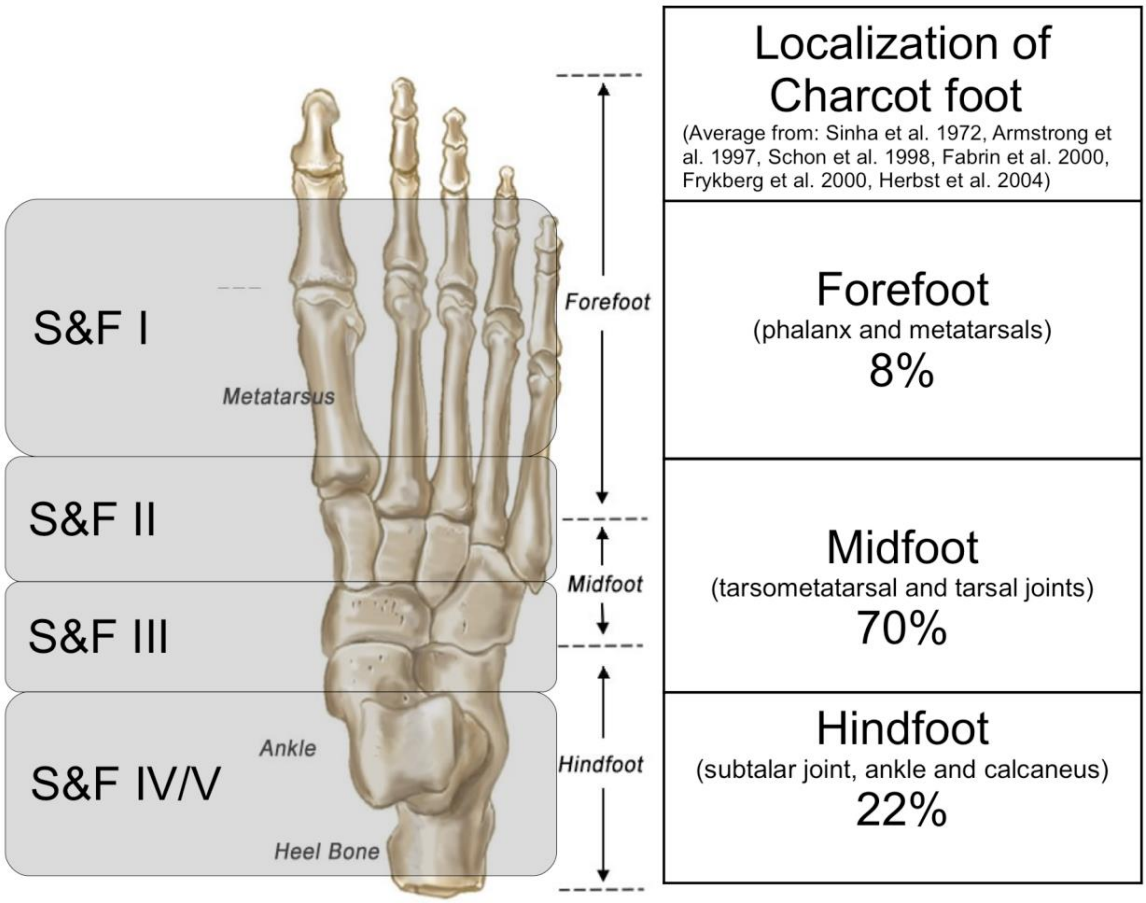


Figure 10 depicts the five patterns of the disease's distribution that can occur alone or in combination for the Sanders and Frykberg 1191 Classification scheme of CF. Specifically, S&F I refers for the forefoot, S&F II for the tarsometatarsal joints, S&F III for the midtarsal and naviculocuneiform joints, S&F IV for the ankle and subtalar joints, and S&F V for the calcaneus [16].

2.1.6 Charcot Foot Deformation in Diabetes: Prevalence, Diagnosis, and Treatment Approaches

Frequently observed in low socioeconomic classes due to subpar living conditions and low incomes, Charcot foot deformity poses significant risks to individuals with diabetes. Smoking is a major contributor to vascular disease and cancer, further increasing the risk. Neuropathy plays a significant role in causing more than half of diabetic foot ulcers, resulting from an uneven distribution of pressure during walking. Contribution factors include inappropriate footwear, poor vision (diabetic cases), and minor major trauma on foot.

Charcot neuropathic osteoarthropathy can lead to anomalies in the middle foot/ankle and subsequent ulceration, often resulting in amputations. Unfortunately, doctors miss diagnosing Charcot foot in about 95% of the cases before consulting a foot specialist [17]. Early detection is crucial, as after 8 weeks, there is a 67% likelihood of problems, including deformities [12]. The prevalence of this disease ranges from 0.1% - 0.9%, emphasizing the importance of early precautions, as it cannot be reversed once it has occurred [12]. Other associated symptoms include old age, declining bone mineral density, rising HbA1c, rheumatoid arthritis, anemia, renal failure, and obesity. Charcot illness really triples the likelihood that a patient will get a foot ulcer, eventually increasing that likelihood to 63% [18].

To study Charcot osteoarthropathy in type 2 diabetes patients, a research methodology was implemented in Pakistan. A total of 1931 subjects of type 2 diabetes, aged between 50 and 60 years, were examined at a specialized diabetes clinic in tertiary care hospital [3]. The assessment included examination was done of erythema, swelling, increase in temperature, and musculoskeletal deformity on both plantar and dorsal surfaces, which were later confirmed through radiographs. Neuropathy assessment involved evaluating pressure, vibration, and joint position. The results indicated that 0.4% of the subjects had Charcot deformity, with 0.05%, 0.15% and 0.2% exhibiting left, right and bilateral deformities, respectively. These findings highlighted the need for special care regarding blood glucose control, early detection, and risk factor management in individuals with diabetes.

Diagnostic imaging plays a crucial role in diagnosing and monitoring Charcot foot deformity. Conventional radiographs have traditionally been used to monitor the disease’s progression and assess bone changes under load, utilizing measurements such as Meary’s angle, cuboid height, calcaneal pitch, and hindfoot-forefoot angle.

The requirement for the automated processing of these pictures is essential given the rising availability and use of medical imaging processes like computed tomography (CT-scan) and magnetic resonance imaging (MRI). To identify CF, it was proposed to identify the best imaging techniques that can be utilized for observation and diagnosis. Traditionally, it has been researched that conventional radiographs have played a vital role as a standard to monitor the spreading of the disease. Radiograph's main role is in assessing the foot bones with each other under load, hence the stated use of meary's angle, cuboid height , calcaneal pitch, and Hindfoot-forefoot angle. Thus, X-rays provide valuable insights for monitoring changes and addressing the issue before it worsens. Magnetic Resonance Imaging (MRI) has also been beneficial in assessing the healing process, success of off-loading treatments, and evaluating soft tissue infections and osteomyelitis. MRI with a large field of view (FoV) is required to capture the spread of the disease throughout the foot. Fluid sensitive sequence like STIR helps assess edema in the bone marrow and soft tissues, while T2- weighted sequence aid in identifying subchondral cysts, identify fluid collections, and sinus tracts [1].

Offloading, through techniques such as complete contact cast (TCC) that cannot be removed, is replaced every 3 days, is a common medical treatment to halt the deformity's growth in acute active stage. and is reviewed every week. Clinical evaluation based on edema, erythema, and changes in skin temperature helps determine the duration and aggressiveness of off-loading [7]. Transitioning to prescribed shoes, boots, or weight-bearing braces is recommended after the acute stage to prevent recurrence or ulceration. Monitoring and antiresorptive therapy, including calcium supplements and oral bisphosphonates, are additional treatment

options. Surgical intervention may be necessary in cases of infected bone, bony prominences, or ineffective adaptations of footwear [7].

2.2 Artificial Intelligence (AI) in healthcare

AI is entering the clinical field as a tool for improving performance [19]. With more data available for healthcare, AI can be used to analyze a variety of pertinent databases with an emphasis on cognitive processes in humans. They include effective artificial intelligence (AI) machine learning techniques including neural networks, deep learning, and support vector machines (SVM). AI can assist medical professionals in making better clinical decisions or replace human-oriented outcomes in disciplines like radiography [19]. AI employs algorithms to gather features, extract data from enormous databases, and utilize that data in clinical settings. To improve feedback accuracy over time, it keeps learning and self-correcting.

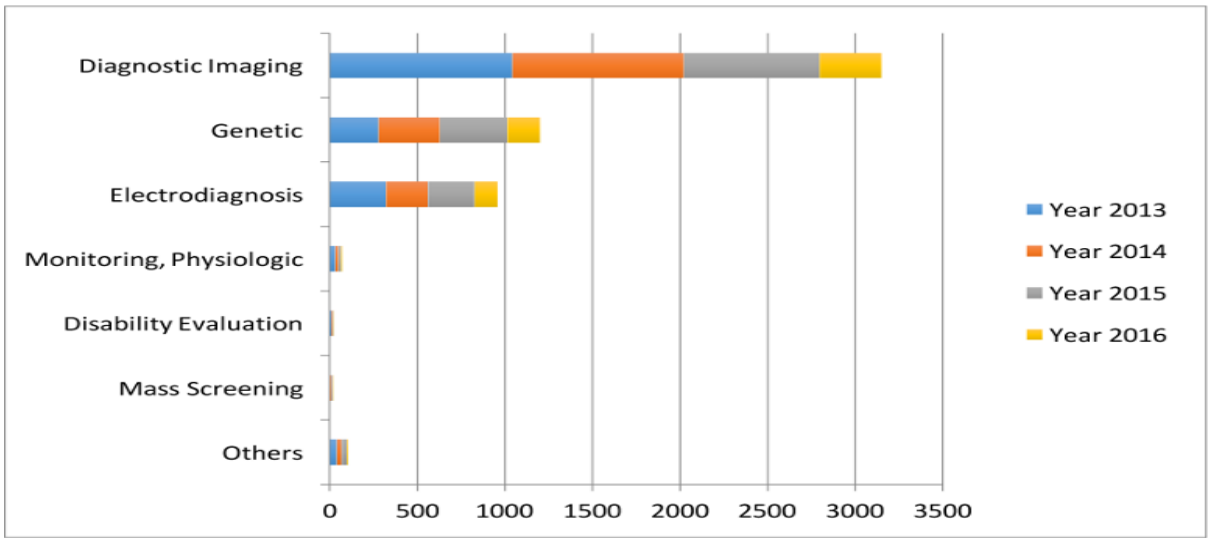


Figure 11 The forms of data considered in the literature on artificial intelligence (AI). The comparison is found by scanning the AI literature for diagnosis methods in the PubMed database [19].

AI is trained using clinical activity data, including diagnosis, screening (X-ray, MRI), demographics, and tests, prior to deployment. They have received instruction in the

pertinent subject results and aspects. Figure 11 shows how much of the AI literature has been dedicated to imaging diagnostics, but current trends in AI adaptation have also demonstrated a major interest in other medical specializations like genetics, electrodiagnosis, and physiology. AI devices fall into two categories, Machine learning and Natural language processing (NLP) [19]. Machine learning analyses structural data from fields like genetics, imaging, and others where attempts are made to group patients or calculate the likelihood of a particular result. To improve the structured data and help machine learning approaches, the NLP method would extract information from unstructured data. As a result, research on using machine learning in the process of medical picture processing has gained popularity. While machine learning tools such as regression and classification tasks are quite popular, Deep neural networks widespread the field of machine learning i.e., regression, classification, segmentation, detection etc.

Segmentation is a widely used technique in image analysis as it offers more advanced classification of images on a pixel-by-pixel basis [20]. Automated image segmentation greatly facilitates the identification and delineation of anatomical features and areas of interest. It plays a crucial role in locating precise points of interest, such as tumors in brain MRIs or skin cancer in dermatological imaging. The effectiveness of image segmentation relies on the quality and quantity of the data being analyzed.

Main reasons for small public datasets:

- Medical data often contains sensitive patient information, making data protection regulations a crucial consideration when sharing public datasets.

- Annotating medical data can be a costly and time-consuming process, which restricts the availability of large-scale annotated datasets.
- Medical conditions and abnormalities are often relatively rare, making it challenging to collect a significant number of diverse cases for public datasets.

Artificial intelligence (AI) finds applications in various fields, including stroke management. Stroke is the leading cause of death in China and affects over 500 million individuals worldwide, resulting in a substantial annual cost to the global economy of approximately US$689 billion [19]. The application of AI in stroke-related studies primarily focuses on early disease detection, diagnosis, treatment, prognosis, and outcome prediction. Systems like IBM Watson, cloud-based CC-Cruiser, and AI Genetic Diagnostic Analysis have been employed to leverage AI technology for healthcare benefits and in-depth clinical research.

2.3 Problem of data availability

Due to privacy concerns, costly and time-consuming annotations, and a scarcity of data samples, obtaining large amounts of data is challenging. The cost of time, patient confidentiality, and annotation were always dependent on the constrained and expensive medical competence due to the difficulties in obtaining funds for study. Success rates are based on the quantity and quality of data that has been annotated by professionals, such as medical specialists. When information is obtained from healthcare facilities for law enforcement purposes, privacy is a key problem. No patient would want a third party to have their health-related data. For example, general data protection regulation 31(GDPR) and personal data act. Asia has its own set of rules such as Personal data protection bill in India [21], The South Korean Personal Information Protection Commission [22] etc.

If research is done under these privacy restrictions, publications are frequently done using just theoretical techniques. Only 30% have used private datasets [23], thus studies are not reproducible. Additionally, the quality of education is impacted by universities and research facilities continuing to use the same medical databases over time.

Costly and time-consuming data annotation is also a big obstacle to tackle for AI algorithms [24]. Pixel-by-pixel annotation is time-consuming, and it is already difficult to locate experts in the subject, even if self-annotation is feasible with the assumption that the data will be evaluated before use in AI algorithms. Thus, it is important to have accurate annotation [25]. Due to extensive annotation use, deep neural networks have demonstrated high grade performance in medical image segmentation.

2.4 Synthetic data

The use of synthetic data could be able to address these problems. It helps with the issues of differential privacy that come up when performing medical research [26]. This type of data can be used by the bulk of ML-using businesses, including robots, agriculture, and health. Some studies employ artificial data to create ground truth. A model called Generative Adversarial Networks (GANs) uses convolutional neural networks and other deep learning methods to create fake images. It is a machine learning unsupervised learning model that automatically detects patterns or regularities in input data to provide results that are comparable to the original dataset. The challenge is framed by GANs utilizing two sub models: a generator model to generate the output and a discriminator model to help distinguish between actual (input data) and false data (generated). Both supervised and unsupervised learning are employed. Both sub models are trained to create output that is

convincing enough to trick the discriminator by producing good examples. A fast-evolving discipline, GANs is assisting the medical industry with its extraordinary capacity to generate essential datasets.

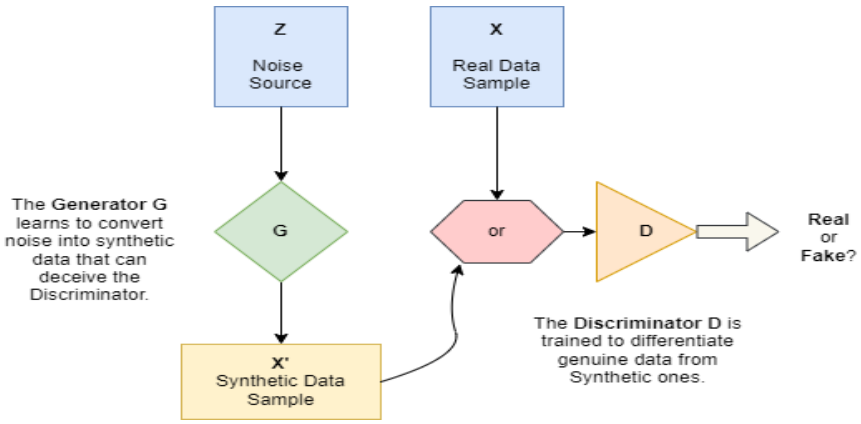


Figure 12 The discriminator D and the generator G are the two models that are taught to a GAN during training. However, they could be implemented by any type of differentiable system that maps data from one space to another. These are commonly implemented using neural networks[27].

2.5 Generative adversarial networks (GANS)

The statistical distribution of training data is used in generative models to produce samples from the learned data. We can do tasks like style transfer, semantic image modification, and data augmentation by synthesizing the data. The GANs structure consists of convolutional layers of fully connected deep networks, spatial filter backs with non-linear postprocessing, and network weights learnt through backward propagation. It deals with multidimensional vectors and italicizes vectors in latent space, or probability space [27].

2.5.1 GAN Architectures

2.5.1.1 Fully connected GANs

On small image datasets, the fully connected neural networks for the generator and discriminator were the first architecture that the GAN developed. i.e., MNIST data, TFD (Toronto Face Data Set) [27].

2.5.1.2 Convolutional GANs

The ground-truth image is itself divided in a Laplacian pyramid, and a conditional, convolutional GAN is trained. This GAN is made up of a Laplacian pyramid of Adversarial Networks (LAPGAN), which decomposes the generation process using a multiscale [27]. To train a pair of stride-based generator and discriminator networks, deep convolutional GAN (DCGAN) was also introduced. DCGAN enables spatial up sampling and down sampling to be learned during training, which is a crucial requirement for mapping the image space to lower dimensional latent space as well as to the discriminator.

2.5.1.3 Conditional GANs

Both the generator and the discriminator have been made class-conditional in the

2-D GAN architecture. The benefit is that it offers better representations for creating multimodal data. By using the fundamental conceptual pipeline, numerous more GANs, including GANs with inference code, Adversarial autoencoders, Vanilla GAN, Cycle GAN, Super Resolution GAN (SRGAN), etc., were produced.

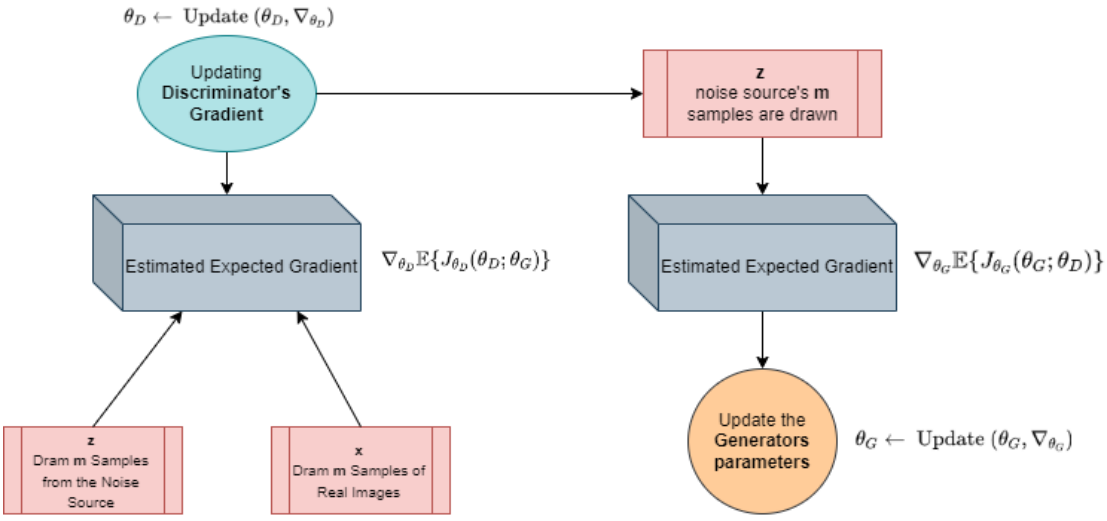


Figure 13 The core GAN training loop. By putting random samples, z, through the generator network, new data samples, xl, can be generated. Prior to updating the generator, the gradient of the discriminator may be modified k times[27].

Semantic image editing, image super resolution, style transfer, classification, etc. are just a few of the uses for GANs.

The Generative Adversarial Network (GAN) is a powerful model that enables the generation of realistic samples without the need for explicit knowledge of the underlying distribution or additional mathematical assumptions. Its versatility makes it applicable to various domains, including both academic and engineering research. In the field of computer vision, GANs have found applications in several areas.

Significant research efforts in GAN development are dedicated to enhancing the quality and quantity of generated images. For instance, the ALPGAN model introduces a convolutional network within the Laplacian pyramid framework, enabling the generation of images in a coarse-to-fine manner [28]. This approach has proven effective in improving the visual fidelity and realism of generated samples.

Self-association GAN (SAGAN) was created, allowing for a broad attention-driven modelling for jobs requiring the creation of images. It is a better version because all the feature maps are used to build high-resolution details. On the ImageNet dataset, it raises the public Inception score from 36.8 to 52.2 and lowers the Fréchet's inception distance from 27.62 to 18.65 [29]. Image Inpainting reconstructs the missing features of an image that an observer cannot tell that it has been restored. This allows us to remove unnecessary details or complete damage parts of old images [30]. Multiple GANs were made each for specific reasons where the other GANs lacked such as in Super-Resolution (upscaling videos) that allows generation of higher resolution images from lower resolution images with realistic detailing [30].

Figure 14 Image Inpainting example [30].

The problem arises that GANs require a large dataset as input to train the model as in data-hungry to train, such as the DCGAN [31]. In cases of a rare disease or a health complex situation of rarity, it would be difficult to train a GAN to reproduce medical data to avoid hinderance of privacy issues.

2.5.1.4 SinGAN-seg GAN

Thus SinGAN-Seg was introduced to train data for medical image segmentation [32]. Its pipeline produces synthetic medical images with corresponding segmentation masks for each output image. This GAN is unique from the others in that it generates a broader set of images from just one single image. It has a basic architecture of SinGAN [33] that has been added with a style-transferring algorithm. The SinGAN may train or learn from a single image. It records the internal patches of distributions (features) and creates qualitative samples that are varied while visually conveying the same information. In this, fully convolutional GANs are arranged in a pyramidal structure, and each one learns patched distributions of the image at various scales. As a result, new samples of any aspect ratio and size are created while generally preserving the same general structure and texture. Once more, the observer is

misled by the produced synthetic visuals when compared to the realistic original image.

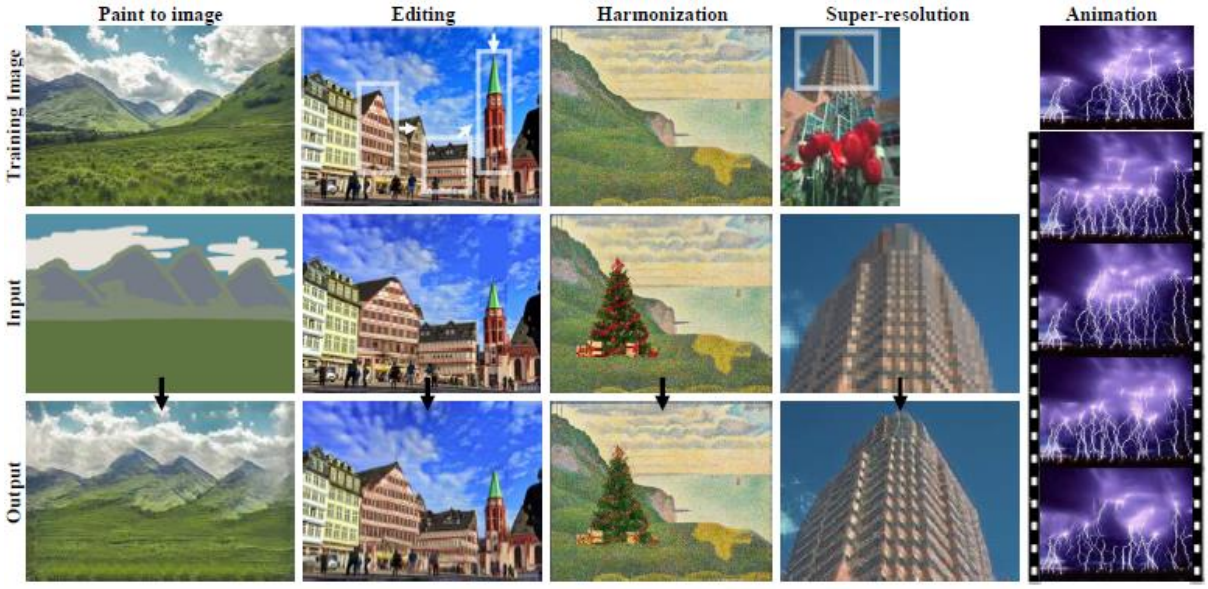


Figure 15 Modification of images. SinGAN can perform a variety of image manipulation tasks, such as converting a paint (clipart) into a realistic photograph, editing, and rearranging the objects in an image, integrating a new object into an existing image, super-resolving images, and producing an animation from a single input. For each of these examples, our model just observes the training image (first row) and is trained the same way across all applications, without any architectural adjustments or additional fine-tuning [33].

It is a brand-new unconditional generative model that is trained on a single image, generating samples only from random input as opposed to a conditional GAN that generates samples based on a "class label," controlling the sort of image that is generated.

There are two main steps imposed in the SinGAN-seg pipeline:

1) Training the generative model per image

2) Transferring style per image.

The synthetic images and accompanying segmented masks would be created in the first step. based on the basic architecture of SinGAN [33] generates multiple synthetic images and masks from one single image and its mask.

Therefore, the generation process identifies as a 1: *N* generation process. Figure 16. Shows this 1:N generation by using $[img]_N$, where *N* represents number of samples generated using the model and the input real image [*img*]. Then afterwards, this is applied in the step for every image in a target dataset, of what we want to generate synthetic data. The second step transfers styles(features) from real images into the corresponding synthetic images. The second step is depicted in Step 2 in Figure 16. This second step is also applied per image.

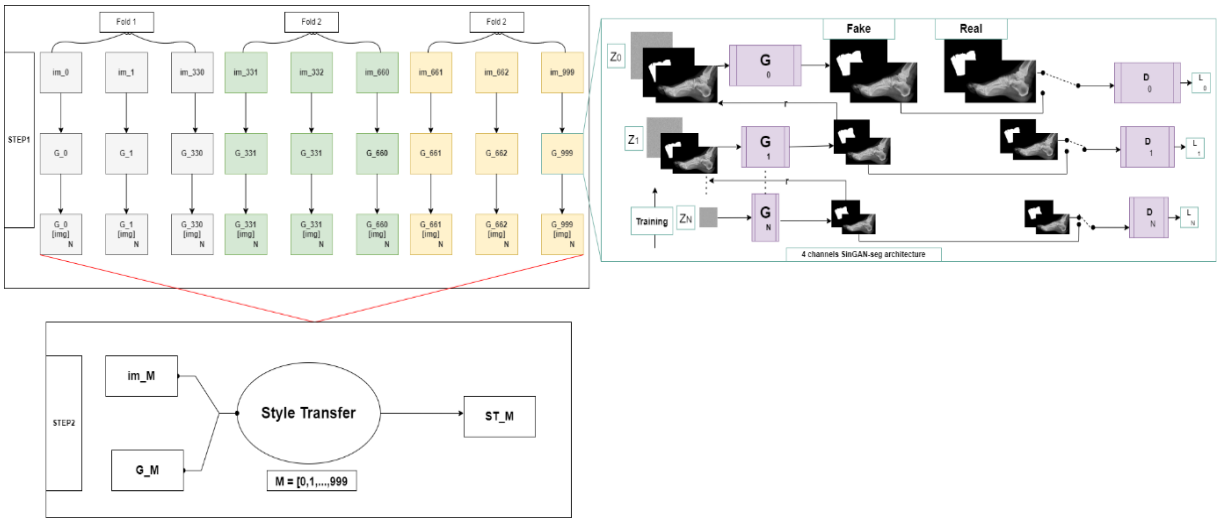


Figure 16 The entire pipeline for training a 4-channel model is shown in step 1. An adjustment employing neural style transfer is shown in Step 2. 4-channels SinGAN is the model's only training stage. Except for GN, each generator in the SinGAN implementation receives a four-channel image (an RGB image and a ground truth) in addition to the input noise vector. The noise vector is the single input for the first generator, GN. Additionally, the discriminators receive four-channel pictures as input, made of a binary mask and an RGB image. The discriminators' inputs may be either true or fraudulent [32].

A single image input can generate synthetic images using the SinGAN-seg model. The SinGAN-seg employs a 4-channel image as input in contrast to the SinGAN, which uses a 3-channel RGB image. It consists of the single-channel ground truth segmented masks and the three-channel RBG picture combined into one image, as seen in fig. Thus, the 4-Channel picture is produced. This makes it possible to create masks for each synthetic image with individual feature adjustments. In the second step of the pipeline, the output image of four-channel goes through style transfer

method for fine tuning, referred to as Neural-style or neural transfer algorithm [34], taking an image and reproduce a new image in a different art of style. It calculates content distance (Dc) and style distance (Ds) to the third image. Contents and styles are transferred using the content: style ratio. This is to improve the quality of generated data by transferring the characteristics of the real images. By Figure 16, we can show every image generated *GM* gets amplified by style transfer from input image *imM*. The produced image is represented by *STM* (M=0,1,2, 3...999). However, this step is optional too, but strongly suggested this style-transferring step to improve quality of the output data from the first step.

Here, we discuss the success that the SinGAN-seg's fundamental algorithm model has had in manipulating visual data. Due to the training of a single image, it has been put up against a variety of algorithms to determine how well it performs.

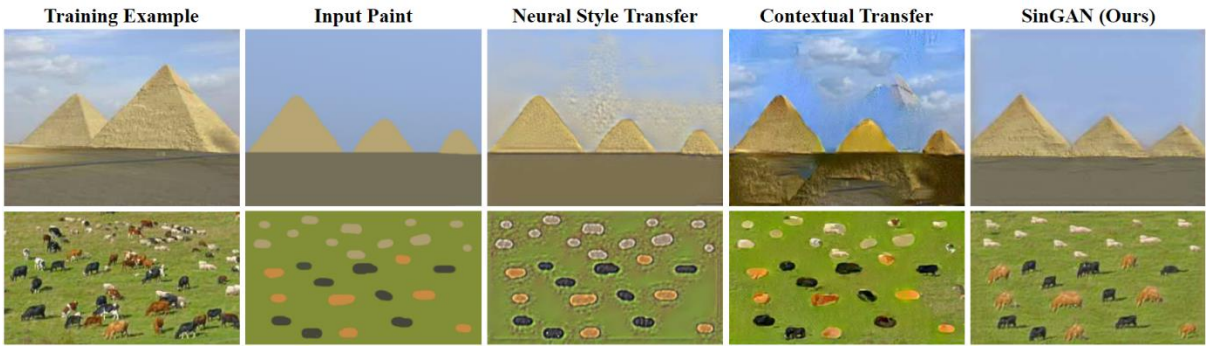


Figure 17 shows how Paint-to-image works comparative to other algorithms. SinGAN was trained on the image and a down sampled version of paint was injected into one of the scales (coarse levels) at test time. The SinGAN generated images preserve the layout and general structure of the clipart while generating realistic texture and fine details [33].

Multiple applications exist due to this model including Paint-to-Image as shown, such as:

- Super-Resolution: A technique to improve resolution of images.
- Harmonization: Blend a pasted object into the background image in a realistic manner.

- Editing: Create a smooth composite with picture sections that have been moved around and pasted in different places.
- Single Image Animation: From a single input image, produce a brief video clip with realistic object motion.

From this scenario, SinGAN-Seg was developed on similar concepts as discussed having applications of producing medical dataset. The polyp dataset that was published with HyperKvasir dataset [reference from Singan-seg paper] was used over which the GAN was implemented. 1000 polyp images with corresponding masks annotated by medical experts.

Using this GAN, a total of 10,000 synthetic polyp images and corresponding masks were generated with high variations when input scale is 0. Under these circumstances while confirming that any segmentation dataset, SinGAN-seg was compared with state-of art deep generative models. Three GAN architectures, DCGAN, Progressive GAN and FastGAN were used for comparison. The resulted synthetic images were compared by Fréchet inception distance which we will discuss later.

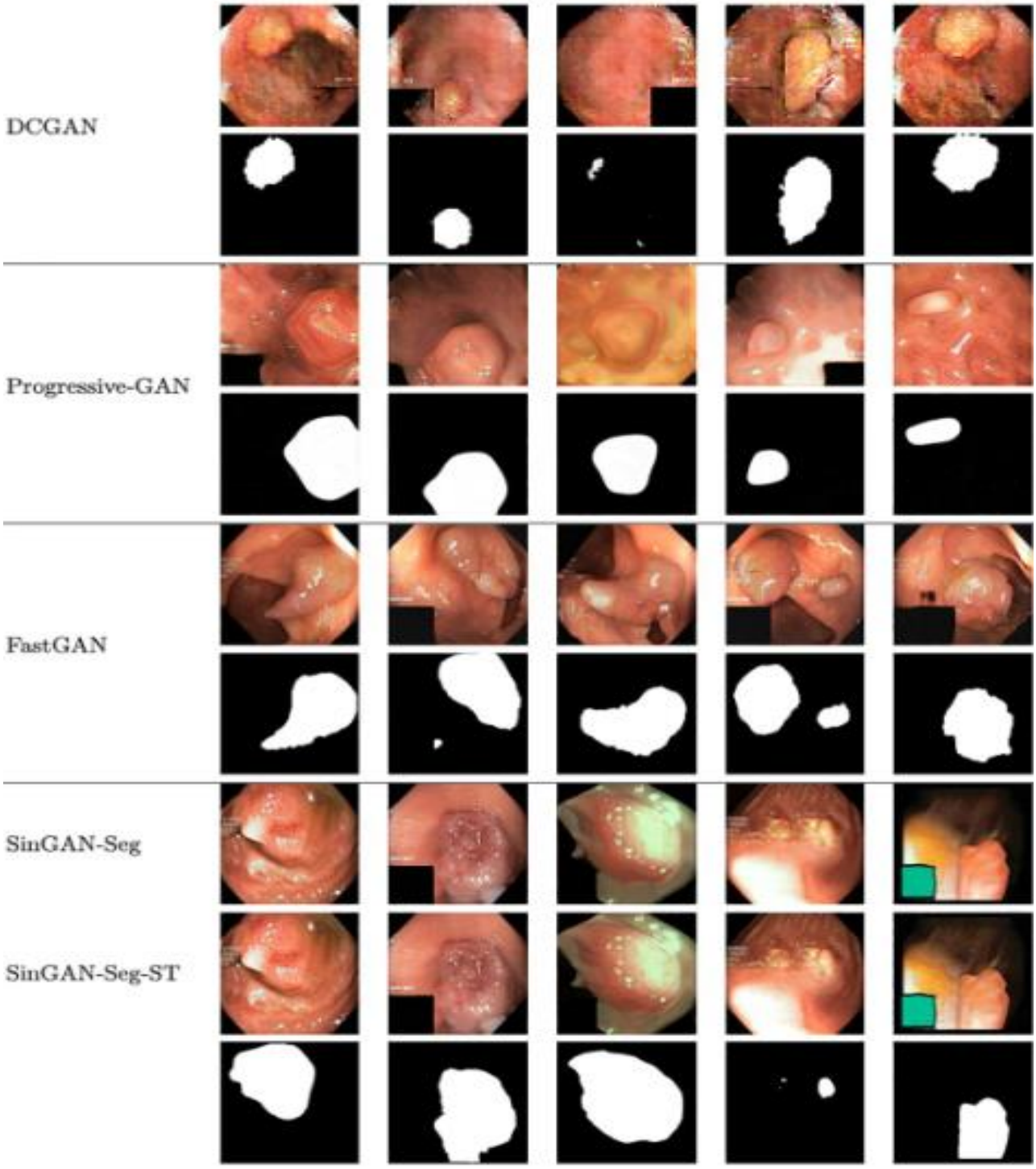


Figure 18 examples of Polyp images produced by various GAN architectures. There are two variations of SinGAN-Seg: SinGAN-Seg and SinGAN-Seg-ST, which both have style transfer. For transferring style, the best ratio of content to style was selected, as 1 to 1000[32].

GAN architecture	Set 1	Set 2	Set 3	Set 4	Set 5	Mean	SD
DCGAN	270.82	269.79	268.38	268.32	269.13	269.29	1.05
Progressive GAN	285.81	284.3	282.81	283.54	285.00	284.29	1.18
FastGAN	74.60	74.43	75.53	75.08	76.20	75.17	0.72
SinGAN-Seg	99.61	98.12	98.27	97.59	97.86	98.29	0.78
SinGAN-Seg with Style Transfer	**43.74**	**43.35**	**43.71**	**43.41**	**43.11**	**43.46**	**0.26**

Table 1 The entire polyp dataset, which contains 1000 images of polyps and the related ground-truth masks, was used to train all GAN architectures. The best checkpoints from each GAN model were then used to create 1000 synthetic images. In SinGAN-Seg, a style transfer ratio of 1:1000 is employed. Bold wording is used to emphasize the best values [32].

Table 1 shows the best score for SinGAN-seg, depicting the fact that the best synthetic images were produced as shown. As a result, the SinGAN-Seg model and training pipeline utilized in this study demonstrate the value of employing them to create synthetic training datasets for GAN models.

Since synthetic data is being considered, it needs to be compared to the real images whether it can be applied for research purposes in as the input data for model training. Thus, evaluation methods are proposed with different metrics to evaluate data. Via [35] we can check evaluation tools that are relevant for our proposed analysis of synthetic data such as object detection competitions, differences, peculiarities, and advantages of widely used annotated formats. Open toolkits are available widely on the internet, performing different matricular calculations to assess the datasets in question. Our ocular vision can distinguish between vibrant colors and depict the object we see through experience, knowledge and learning that

includes the structure, distance comparisons etc. Deep Learning convolutional networks has become a methodology of choice for medical image analysis [36]. In case of Machine learning, it has provided us with Algorithms that would make it easier to detect a specific object by model training to aid a person in detection of features in an image. Recently with the involvement of AI and machine learning to assist research, the results are often described with datasets and metrics. This has increased the demand for annotated data, specifically in the medical field. Annotation tools are being widely used in aiding to check similarities or identify the objects in case such as images [35]. Manual annotation requires expensive and time costing methods in which an expert is asked to annotate the data. Evidently speaking, medical schools/facilities are also moving towards competency-based education and research thus require trainee skills without human intervention [37]. Recent studies have also shown evaluating metrics in image segmentation lacks model performance assessment reliability and statistical bias when metrics are implemented incorrectly [38]. The IOU (Intersection over Union) has been a widely used 2D/3D object detection and semantic/instance segmentation metric for evaluation. The combination of Regression and Classification has been defined in combination in the Deep Learning Algorithms. Considering the case of Object Detection, regression is used for the bounding box whereas classification is used for the class inside the bounding box.

2.6 Machine learning libraries

Machine Learning libraries are software packages providing a collection of algorithms, tools and functions that can be implemented for developing machine learning models. These libraries help in simplifying training models through data processing utilities, pre-implemented algorithms, and evaluation metrics.

These packages support the advancement of machine learning techniques and act as aiding to simplify complexity of algorithms implemented from scratch, making it easier for practitioners and researchers to experiment on different datasets and models.

These libraries are kept to the latest updates that continue to improve and evolve, pushing the boundaries of machine learning to explore new computational areas.

Keras

TensorFlow is a google developed library for machine learning applications in deep learning. It can be run on CPUs, GPUs and IoT processors. Via TensorFlow, that basically consists of multidimensional arrays called tensors produce a library through which it is run is called Keras, a neural network API based on Python [39]. It has four working principles:

- User friendliness
- Modularity
- Easy extensibility
- Work with Python

Data structures in Keras have the most important aspect called models. There exists multiple type of Keras layers available for configuring in terms of neural network as listed below:

- **Dense:** Core layer in densely connected NN layer.
- **Activation:** A function that is applied to a given input
- **Dropout:** Input layer given with transformation of dropout.

- **Flatten:** for flattening the input without affect to size of both input and output and used in corresponding to model of data tested and trained.
- **Reshape:** converting the shape of input tensor to output tensor.
- **Permute:** to change dimensions of input and for acts relative to connecting convolution networks (ConvNets) and recurrent neural networks (RNN).
- **Lambda:** wrap arbitrary expression as layer objects.
- **Activity Regularization:** apply update to cost function that is based on input activity.
- **RepeatVector:** allows to repeat input vector **n** times.

There are also other core layers similar like spatial dropout and masking [39].

1. **Convolutional layers:** Like TensorFlow, convolutional layers available in Keras consist of Conv 1D, Conv 2D, Separable Conv 1D, Separable Conv 2D, Depthwise Conv 1D, Depthwise Conv 2D, Conv 2D Transpose, and 3D.
2. **Pooling Layers:** It includes average pooling, max pooling, average max pooling, and global max pooling.
3. **Locally Connected Layers:** they include both locally connected 1D and 2D layers.
4. **Embedding Layers**
5. **Merge Layers**
6. **Advanced Activation Layers:** Developed for special cases such as LeakyRELU (rectified linear unit), PReLU (parametric variation of RELU), Exponential linear unit (ELU) and threshold ReLU in addition to SoftMax are also available.
7. **Recurrent Layers:** functional exclusive with recurrent neural networks (RNN), long short-term memory (LSTM) layer and gated recurrent unit (GRU).

8. **Normalization Layers:** normalization layer (Batch Normalization) provided in Keras.
9. **Noise Layers:** To reduce noise in data provided such as alpha dropout and Gaussian noise etc.
10. **Layer Wrappers:** Time distributed wrapper that is applied to the input of time space and bidirectional wrapper used with RNNs.

Segmenting an image involves dividing it into various classes of pixels. This technique has various applications, such as identifying tumors medical images and differentiating between land and water areas in drone images. Unlike image classification, that uses CNNs to output probability score vector, image segmentation requires CNNs to produce an image. To achieve this, traditional CNN architectures are typically modified. There are many architectures available for image segmentation, including transformers. Researchers are also exploring other methods to improve performance, such as the use of deep-supervision loss.

Image segmentation involves partitioning the pixels of an image into distinct classes. Applications of this technique include identifying tumor regions in medical images and separating land and water areas in drone images. Unlike image classification, where CNNs output a probability score vector, image segmentation necessitates CNNs to output an image. To accomplish this, conventional CNN architectures are often adjusted. There are various architectures available for image segmentation, including transformer-based models. Additionally, researchers are continually experimenting with other techniques to enhance performance, such as the use of deep-supervision loss.

Traditionally, CNNs are adept at recognizing the presence of objects in an image. However, for segmentation, CNNs must also learn to accurately locate these objects within the image. The U-Net architecture is designed specifically for this purpose. The original U-Net paper describes it as a network consisting of two parts: an encoder section and a decoder section.

2.7 U-net

For a deep network to be successfully trained requires sample data annotated in thousands of data. Thus, a strong use of data augmentation is required so annotated samples are trained for classification efficiently. In the past few years, deep convolutional networks have made a name in tasks of visual recognition. However, there were limitations because of the size of networks and training datasets. That breakthrough was accomplished by ImageNet with parameters for one million training images over a large eight-layer network [40]. Convolutional networks were based for tasks to classify where a single class label is the output of an image. However, in the case of medical images, localization was in demand where each pixel is assigned a desired label. Thus, a network was trained to predict each pixel class with a patch (local region) around the pixel. This turned out to be the U-net as shown with an architecture [41].

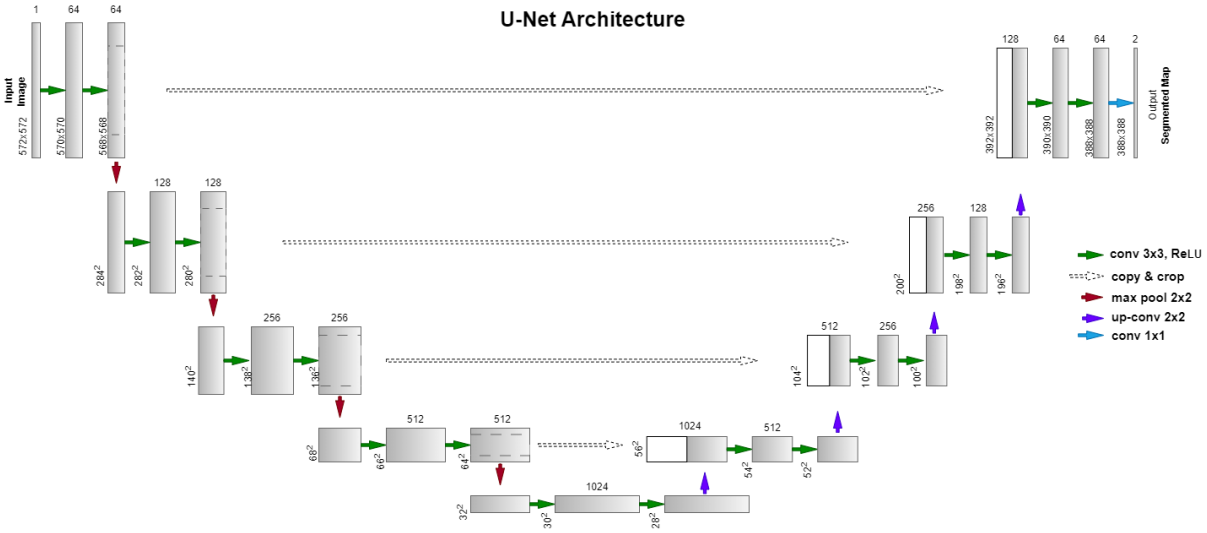


Figure 19 As shown in Figure, the U-net architecture (used for an example of 32x32 pixels in the lowest resolution) utilizes blue boxes to represent multi-channel feature maps. The number of channels is indicated at the top of each box and the x-y size can be found at the lower left edge. White boxes symbolize duplicated feature maps and arrows indicate the different operations used within the architecture.

The U-net architecture is characterized by its “U” shape, which is symmetrical in design. It is compared of two main sections: the contracting path on the left and the expansive path on the right. The contracting path is made up of convolutional layers that reduce the spatial dimensions of the feature maps. The expansive path, on the other hand, is composed of transposed 2D convolutional layers which increase the spatial dimensions of the feature maps. This approach can be thought of as an up-sampling technique. This network is also termed to be a ‘fully convolutional network” [42]. The architecture of the U-net can be explained part by part:

Contracting path

```
conv_layer1 -> conv_layer2 -> max_pooling -> dropout(optional)
```

Each process in this path is made up of two convolutional layers, with the number of channels changing from 1 to 64. This convolutional process increases the depth of image. The red arrow pointing down in Figure 19 represents the max pooling process, which reduces the size of the image by half (e.g., from 572x572 to

568x568). The process is repeated three times until it reaches the bottommost. Two convolutional layers are built without the need of max pooling.

Expansive Path

```
conv_2d_transpose -> concatenate -> conv_layer1 -> conv_layer2
```

This is the path through which the image is upsized to the original size of input. The expansive path of U-net architecture uses transposed convolution, which is up sampling technique that increases the size of images. It works by adding padding to the original image, followed by a convolutional operation. As depicted in the architecture Figure 19. the image size is increased. This image is combined with corresponding image from contracting path to incorporate information from previous layers for more accurate predictions. This process is also repeated three times, reaching to the uppermost part of the architecture where the last step is reshaping the image to requirements for prediction. The last layer consists of no dense layer and just a 1x1 filter sized convolutional layer.

Thus, in conclusion, U-net can do image localization by predicting the image pixel by pixel and claimed in the paper [41] that the network is strong to predict with few datasets. Many applications for image segmentation are using U-net.

There are multiple types of U-net present that were introduced by Keras. The project under the name of Keras-U-net-collection consists of functions for configuring Keras models with user friendly hyper-parameter options that include hidden layer activation, network depth and batch normalization for all U-net variants. These variants are as follows:

2.7.1 U-Net plus

The proposed architecture follows the fully convolutional network (FCN) approach and is designed as an end-to-end network. It leverages convolutional operations to extract features from the input image. In the U-Net plus architecture, all layers except the final layer utilize 64 channels in their convolutional operations and are connected to a rectified linear unit (ReLU) activation function. The final layer of the U-Net plus architecture employs a 1x1 convolutional size with two channels. This configuration is employed to map the feature maps to the desired number of output classes. The inclusion of 1x1 convolution in the final layer can enhance the network's robustness, which is a common practice in deep neural network [43].

The architecture can be illustrated in following diagram:

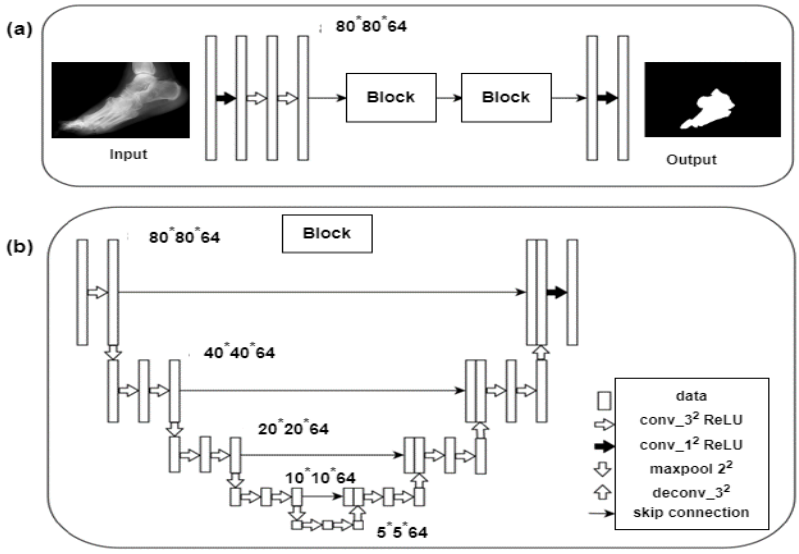


Figure 20 The U-Net Plus architecture is composed of two main parts: the main architecture and the detailed architecture of a block. The main architecture (a) is the overall structure of the network, which incorporates multiple blocks to extract features from the input image. The detailed architecture of a block (b) includes various operations such as convolution (conv), rectified linear unit (ReLU) and skips connections.

The architecture utilizes skip connections, which link the corresponding feature maps from the down sampling and up-sampling stages. This helps to alleviate the problem of spatial information loss that can occur during down-sampling and improves the network's ability to focus on extracting useful features through convolutional

operations. This architecture utilizes skip connections, which link corresponding feature maps from the down-sampling and up-sampling stages. This helps to alleviate the problem of spatial information loss that can occur during down-sampling and improves the network's ability to focus on extracting useful features through convolutional operations.

A block in the network architecture includes a skip connection, as well as other operations such as convolution, pooling, and deconvolution. Despite the multiple feature extraction operations performed within a block, the feature extraction process following the skip connection must be based on the previous input information obtained through the skip connection. The skip connection and the operations between it can be thought of as one skip connection operation. This operation extracts the features of the input information before it can be used as input for the next operation. The skip connection combines feature information from both small and large field of views, which allows for the use of global features. However, the feature maps from small field of views are from early features in network. A block can be thought of as a skip connection operation that is like convolution, but with stronger feature extraction capabilities. By using two blocks in succession, the network can enhance its ability to extract complex abstract information.

2.7.2 Attention U-Net

Fully Convolutional Networks (FCNs) have been shown to achieve superior performance compared to other commonly used registration-based methods in image segmentation. U-Net architectures allow effective computing of segmentation masks at the same resolution as input images. The fully convolutional architecture also allows for the use of images with varying resolutions, unlike traditional convolutional networks which have layers that are dependent on the input size. In

medical image analysis tasks, the integration of attention Gates (AGs) into the U-Net architecture has been shown to improve the accuracy of segmentation models while maintaining computational efficiency. The proposed Attention U-Net architecture is illustrated in figure 21. Traditionally, cascade CNNs were used to select the Region of Interest (ROI) where the target organ is likely located, without using Attention Gates (AGs). However, by incorporating AGs into the Attention U-Net architecture, it eliminates the need for a pre-selection network. Instead, the Attention U-Net learns to focus on the most important local features, while downplaying the less relevant ones. This also results in a decrease in false positive rates [44]. Attention Gates (AGs) are commonly used in a variety of applications, such as natural image analysis, knowledge graphs, and natural language processing (NLP) tasks such as image captioning[45] machine translation and classification task.

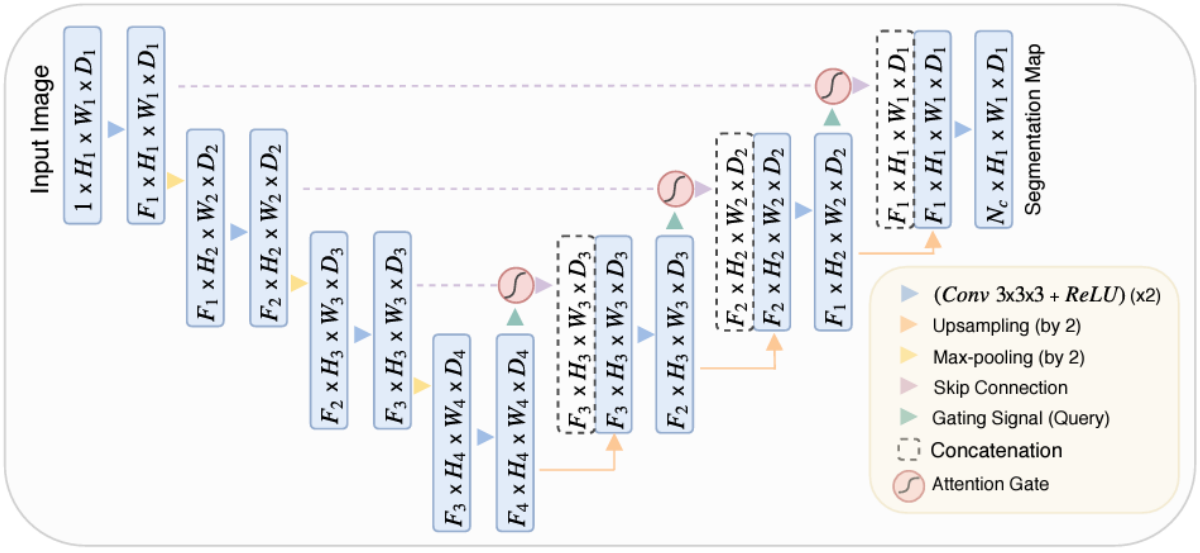


Figure 21 The proposed Attention U-Net segmentation model consists of an encoding part, where the input image is progressively filtered and down-sampled by a factor of 2 at each scale. The number of classes is denoted as Nc. Attention gates (AGs) are utilized to filter the features that are propagated through the skip connections [46].

2.7.3 Recurrent and Residual U-Net (RU-net & R2-Unet)

Two new models named RU-Net, and R2U-Net were introduced for medical image segmentation.

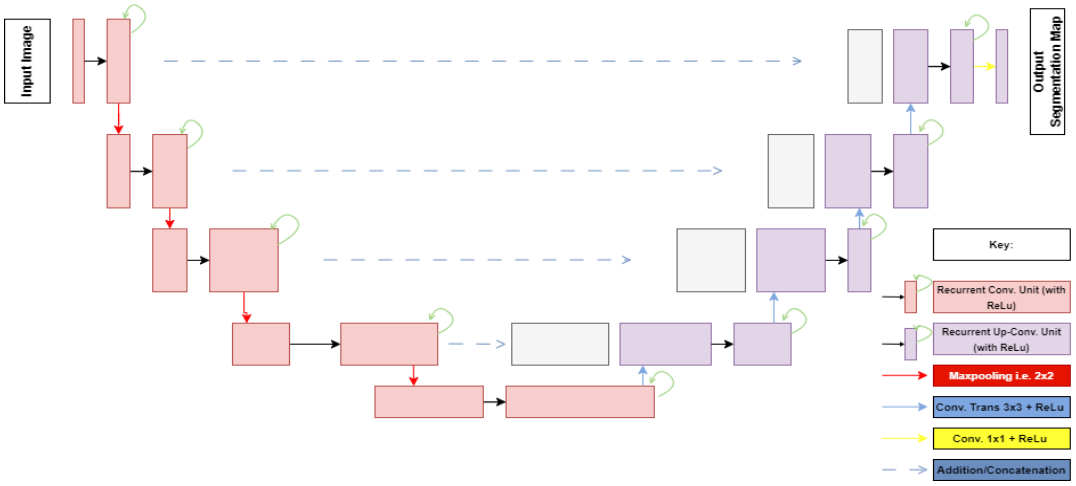


Figure 22 The RU-Net model is an architecture that incorporates convolutional encoding and decoding units with recurrent convolutional layers (RCLs) based on the U-Net architecture. While the R2U-Net model is an architecture that uses residual units in conjunction with RCLs [47].

These two new models, RU-Net and R2U-Net, combine the strengths of three recently developed deep learning models: U-Net, RCNN and its variants. The RU-Net model is based on the U-Net architecture, with the addition of recurrent convolutions before down-sampling, before up-sampling, and before the output of the segmentation map. The R2U-Net model is like RU-Net, but it utilizes residual learning instead.

U-Net and its variants have been used in multiple scenarios to achieve the segmentation for clinical images to study/diagnose health issues. For example, Study has been carried out using U-Net plus for semantic segmentation for esophagus and esophageal Cancer in computed tomography Images. Doctors can diagnose esophageal cancer with the use of accurate segmentation and 3-D representation of the oesophagus and esophageal cancer from computed tomography (CT) images. The segmentation of the oesophagus and esophageal cancer is quite challenging due to irregular and ambiguous boundaries. A 2-D CT slice could be used to segment the oesophagus and esophageal cancer using U-Net Plus. Two blocks are used in this network architecture to improve the efficiency of

feature extraction from complex abstract information, which can successfully resolve irregular and ambiguous boundaries. A skip connection operation known as a block is comparable to convolution. The architecture is tested with 295 slices from 6 CT scans after being trained on a dataset of 1924 slices from 10 CT images. To model segmenting the 3-D CT image, the training and test datasets are each multiplied by ten. It reported a 0.790.20 dice value and a 5.879.91 Hausdorff distance using the new framework. The 3-D segmentation of the oesophagus or the detection of esophageal cancer is then constructed using a semi-automatic method. To aid in the diagnosis of esophageal cancer, 3-D renderings of the oesophagus or esophageal cancer are used [43].

In another instance, Attention U-Net was tested for Chest X-ray Lung Segmentation. The most prevalent test among medical imaging modalities is the chest X-ray (CXR). It is used for the detection and distinction of a variety of diseases, including pneumonia, lung cancer, tuberculosis, and COVID-19. The workload of the physicians is significantly reduced while reliability and quantitative analysis are improved when computer-aided detection methods are integrated into the diagnostic pipeline for radiologists. Modern fully convolutional neural networks and an adversarial critic model are combined in our approach. On the Japanese Society of Radiological Technology (JSRT) dataset, it achieved a final dice score (dice coefficient) of 97.5% and generalized well to CXR images of unseen datasets with different patient profiles [44].

The extension of U-Net architecture models called RU-Net, and R2U-Net were also evaluated in three different applications of field of medical imaging that included retina blood vessel segmentation, skin cancer lesion segmentation and lung segmentation. Their experimental results showed better performance in

segmentation tasks with same number of network parameters when comparing with existing methods [47].

2.8 Intersection over union (IOU)

Image segmentation is a subfield of image processing within computer science. It is used for identifying common features in input image by learning and labelling pixels with class i.e., kidney, brain tumor etc. wide range of algorithms are available for solving these segmentation problems but by the introduction of convolutional neural networks and deep learning, it has an extensive use presently [48].

It's quite difficult to implement this in real life situations, thus the IOU comes into play. The concept of IOU is to use it as a metric in evaluation of deep learning algorithms that estimates the predicted mask (in medical terms) matches the ground truth. It is also referred to as Jaccard Index.

To use the IOU metric, the following are required:

- Annotated ground truth (bounding boxes or masks).
- Similarity of model's prediction as ground truths.

IOU Formulae

Intersection over Union is an instinctive metric, thus not many challenges are faced when understanding. It is calculated by the ratio of predicted and ground truth annotation over union of these.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Figure 23 The formulae for Jaccard index

The formulae can be visualized with bounding boxes, such as

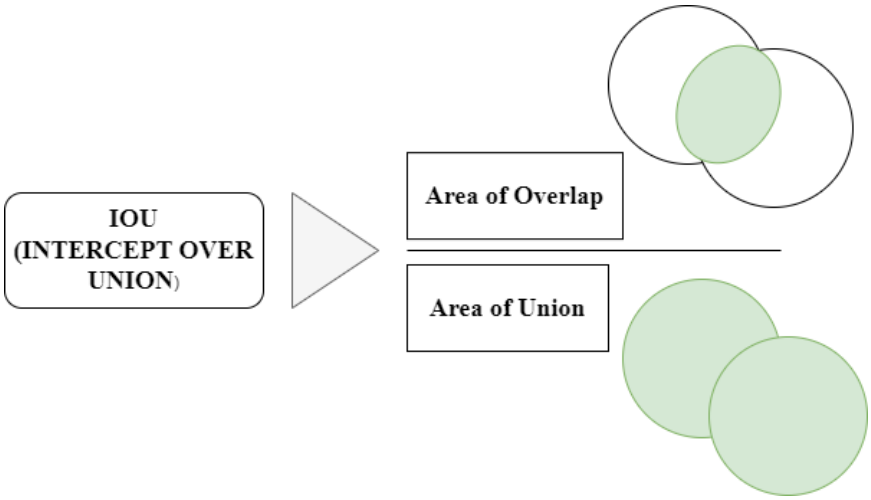


Figure 24 IOU Equation [48]

Thus, this shows to be used as a reasonable metric for model to make predictions heavily overlapping the ground truth.

The IOU algorithm runs such that:

1. From your model, get the prediction.
2. Make a comparison of the mask (predicted bounding box) and the ground truth.
3. Calculation of the overlap area and union between the two areas.
4. Divide the overlap value with the union of these areas.
5. Make analysis of the result.
6. Repeat steps 1-5 for all the test image dataset.
7. Get the average IOU scores for the result.

The higher the IOU score, the better results (higher overlapping). The greatest value is 1 while the least being a 0. On the conditions, we can set our own thresholds for evaluation.

With modern imaging and its increased availability made a need for automated medical image segmentation.

(IOU) metric has found applications in various contexts, including object detection and semantic segmentation. For instance, IOU analysis has been used to evaluate the performance of image detection/segmentation models based on (CNN). These neural networks are capable of extracting complex hierarchical structures of object structures of objects from large datasets. Through extensive training on abundant data, deep learning systems can acquire the ability to recognize and interpret images, exemplifying a key characteristic of artificial intelligence. Two categories of deep learning algorithms for image interpretation can be made. Examples of completely convolutional algorithms that employ an encoder-decoder architecture include SegNet, U-Net, and SharpMask. In contrast, region-based approaches like Mask-RCNN, PSP Net, and DeepLab extract features using a stack of convolutional neural networks (CNNs). For most picture segmentation tasks, fully convolutional techniques outperform region-based methods when the networks are trained on enough annotated datasets. One design was produced that used deep learning methods based on the Mask-RCNN to recognize 2D images while segmenting each mask component as part of a study [48].

2.9 Research gap

To develop models that would aid in the detection of illness and health issues for research, enhancing life and saving lives, society now needs machine learning and AI. In general, bone fractures and cancer have been identified using machine learning techniques. However, in our scenario, we will be able to detect the advancement of bone destruction for CF disease, which is difficult to diagnose in a diabetic condition. To diagnose with the aid of artificial intelligence methods,

additional bone damage must be stopped. This will stop any future delays and lessen the possibility of a higher health decline.

To identify Charcot foot disease in the bones using medical imaging, our research will fill a gap in the literature by combining AI and machine learning. A significant portion of research on CF disease focuses on its study of progression, including how medical imaging is a fantastic approach for monitoring and diagnosis, how early discovery is advantageous, and how delay in detection and progression can aggravate the patient's health. The development of a detection model that may be used for early prognosis is still pending among experts. To increase the amount of data available for training and creating segmentation models for certain diseases and health issues, GANs are being utilized to make medical images. They are only tested on datasets that are widely accessible, though. These models need at least 10,000 photos to be trained on powerful GPUs, hence little research has been done based on the availability of these resources. Additionally, single image based GANs continue to be an underappreciated product because the medical field has not utilized them extensively. Furthermore, because CF is a rare, underdiagnosed condition, no machine learning or AI techniques have been used in this case yet. Only experts are currently addressing this case using their own observations and the cutting-edge medical equipment found in hospitals.

3 METHODOLOGY

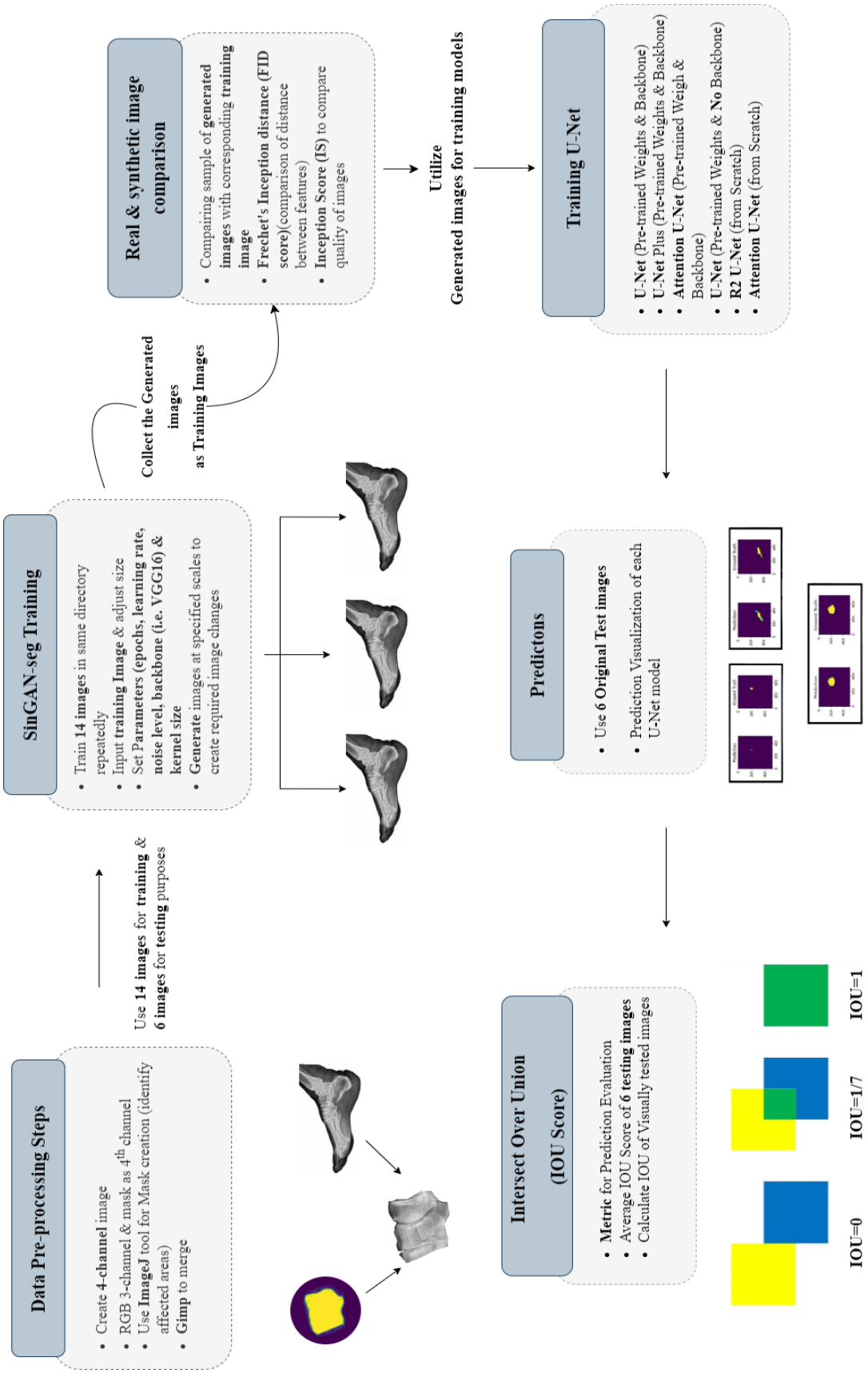


Figure 25 Methodology Flowchart

This section displays the model that was trained to create the artificial images, which is followed by an open-source visual object detection interface to compare the artificial data with the real data [35] after which training a model to recognize the diabetic CF utilizing the total data generated.

3.1 Data Collecting and Processing

In the preliminary stage, we collected a dataset of medical images from a healthcare facility in Pakistan, with the necessary permissions of the local health departments. Our aim was to investigate the prevalence of diabetic feet in the country and understand the emerging challenges related to diabetic-related health conditions. Given that foot ulcers are a symptom of developing Charcot foot in the future, a study to determine the prevalence of foot ulcers and factors related to diabetes was undertaken in the Pakistani state of Punjab. Between 2018 and 2019, a sample of roughly 1503 diabetics was gathered. A multiple logistic regression was performed among those over the age of 18, and the total prevalence of foot ulcers was shown to be 16.83% (95% CI: 14.9 - 18.7%). Whereas the prevalence for females was 17.52% (95% CI: 15.2 - 19.9%) and prevalence for males was 15.48% (95% CI: 12.3- 18.6%)[49]. Contrast to rural areas, where prevalence was 13.91% (95% CI: 10.6 - 17.2%), urban areas had a prevalence of 17.96% (95% of CI: 15.7 - 20.2%)[47]. They had the highest prevalence for people aged 75 and older, at 66.67% (95% CI: 51.9 - 81.5%). Overweight persons had a prevalence of 25.49% (95% CI: 21.3 - 29.7%) and those with incomes over Rs. 61,000 had a prevalence of 24.4% (95% CI: 15.8 - 32.7%), respectively [49].Therefore, evidence from this study indicates that foot ulcers are very common in Punjab, Pakistan. To obtain the medical photos, contacts have been made with numerous health institutions. It is fair to note that medical image processing is rarely used to identify CF across numerous

visits. Most patients are unable to pay for the pricey testing process. To stop the disease from spreading, diabetologists, doctors, and orthopedic surgeons directly use surgical techniques. Patients are commonly not advised to have medical imaging diagnosis, so data in the form of MRI images and CT scans was infrequently available. However, because X-rays are so inexpensive, patients are encouraged to undergo this imaging diagnosis. Unfortunately, hospitals in Pakistan still use paper documentation because much of it has not yet been digitalized. Many hospitals do not grant access to such information, even though patients' privacy is protected, and security access is restricted to data for ethical grounds. To obtain the dataset, which is a lengthy procedure, one would need to go via the proper network, i.e., a university, for academic purposes while simultaneously giving agreement that the data won't be utilized for other purposes. With time, medical information, including X-rays of patients who were recently diagnosed with early CF stage by an orthopedic, was submitted with the patients' complete consent. Early stage of CF was identified in the medical information because patients took the appropriate measures before Stage 3 symptoms developed. We can utilize this to determine the early stages of the disease. A total of 20 images, showing the x-ray scan from various angles for 4 patients receiving medical treatment for CF disease, were available. With a few X-rays depicting the local foot area affected by CF removed from the forefoot side without significant bone loss, the images of the CF revealed the bones fractured from the mid and higher footing.

Figure 26 X-ray Images of CF

Patient	Gender	Profile view X-ray	Posteroanterior X-ray	Total
A	Male	2	2	4
B	Female	4	6	10
C	Female	1	1	4
D	Male	1	1	2
				20

Table 2 shows the x-ray image dataset contributions of 4 patients having the CF in early stages.

Synthetic data generation presents a potential solution to address the limited availability of data, as the provided data was insufficient for training a machine learning model. In this case, synthetic data is created using X-ray images in JPEG format. X-ray images are considered the most effective method for diagnosing deformities and study the bone regression of CF.

Dimensions	Width	Height	Horizontal Resolution	Vertical Resolution	Bit depth	Size
970 x 1600	970 pixels	1600 pixels	96 dpi	96 dpi	24	245 KB

Table 3 shows details of X-ray images collected.

The following prerequisites for the GAN input requirements must be completed before attempting to generate synthetic data:

1. Convert format from jpeg to png.
2. Use open-source tool to annotate for extracting masks.
3. Add the layers of the test X-ray and its corresponding masks together to create a 4-channel image (3 channels for RGB and the 4th channel is the mask).

Then, using the annotating tool, we create masks. At the National Institutes of Health and Laboratory for Optical and Computational Instruments, ImageJ is a Java-based image processing application [50]. 8-bit, 16-bit, and 32-bit images can be shown, examined, edited, saved, processed, and printed using it. JPEG, DICOM, FITS, GIF, BMP, PNG, and raw are among the image formats that can be read by the tool. Additionally, it enables "stacks," which are groups of photos that share a single window. With the help of its multithread software, laborious tasks can be combined with simpler ones. Its open architecture was intended to allow for extension using Java plugins. The tool's built-in editor and Java compiler can be used to create analysis, unique acquisition, and processing plugins. The User-written plugins solve any image processing and analysis problem. ImageJ was used to trim where necessary, annotate the X-ray images to construct the masks (created with the orthopedics observation), reduce to a workable image size to avoid overfitting, and convert our image format from Grayscale to RGB image. To improve accuracy, we magnify in on the images and place precise marks on the foot structures that are labelled as being afflicted by Charcot foot disease using the polygon selection tool. Once everything is finished, we save the RGB X-ray image together with the associated grayscale mask as PNG files.

Gimp 2.0 [51] is used to create a 4-channel image utilizing the X-ray image layer and the associated mask layer. A free open-source alternative to Adobe Photoshop for raster graph editing is called Gimp [51]. It is a cross-platform image editor that works with Windows, macOS, Linux, and other operating systems. Both third-party plugins and a variety of customization options are supported. It has graphic design aspects with assisting plugins like programming methods, extension & flexibility etc. It can modify high quality images, produce original artwork, and comprise of graphic design elements.

The four layers make up the four-channel image. The Red, Green, and Blue (RGB) channels combine to create an image of color combinations in the first three levels, which are the RGB layers. The fourth channel is the alpha layer, which controls what is seen in the image and transparently delete any superfluous portions. Opening the associated mask and the CF X-ray image is the first step in the procedure. We add a layer mask and an alpha channel to the image. Adding a black, white, or grayscale layer is not required. Onto the mask layer in the alpha channel, we copy and paste the mask image. After producing the alpha channel with the matching mask and merging the layers into a single image, we have a 4-channel CF X-ray image.

3.2 Synthetic X-ray Image Production Using SinGAN-seg.

We propose the idea of implementing the SinGAN-seg after analyzing the issues relating to privacy, dataset availability, and to address the problem of time-consuming and expensive expert annotation [31] to create numerous synthetic photos to help us train a machine to recognize CF when confronted with a sample patient medical image. By the concept of U-Net++ [52] applied on the SinGAN-seg, Features are extracted by segmenting the images to create similar real like output of images.

3.2.1 Training the generator

SinGAN-seg requires parallel programming to create synthetic images because it is a GPU-intensive model. In contrast to SinGAN (Original), which just uses an RGB image with a noise vector and outputs synthetic graphics, it consists of layered input and output.

All the generators in this model (G0 to GN-1) receive a 4-channel picture as input along with a noise vector. The discriminator receives the same input as the generator, which can be real or fake, and works simultaneously with the generator in the first step, only accepting a noise vector as input. With a single image as input, this GAN creates randomized synthetic images with varying alterations according to scale. The GAN uses a pyramidal training structure to extract features at various scales between 0 and 9 using various kernel sizes. Depending on how much change we require, we can generate images at various scales.

Since this GAN is built on the SinGAN architecture, learning an unconditional model that captures the intrinsic statistics of the image is the stated goal. The training samples are taken as patches rather than the entire image according to the settings of standard GAN [33]. It serves to capture the intricate visual structure at various scales. This enables us to capture the overall property, such as the image's form and fine texture details. This is illustrated in Figure 16. (SinGAN architecture) with a hierarchy of patch-GANs called Markovian discriminator. Each patch oversees capturing distributions at various scales. They can't memorize the entire image due to their narrow receptive fields and capacity. As a result, internal learning is done on a single image different from the perspective of other GANs. The pyramid shows generators stacked together {G0,…,GN), trained against image pyramid x:{x0,…,xn} where each image is a down-sampled version of x by factor r^n, for r>1. Each

generator Gn produces realistic image samples corresponding to the patch distribution with images achieved by adversarial training, where Gn fools the associated discriminator Dn that tries to distinguish the patches samples with patches of xn.

It starts from the crudest scale and passes sequentially through the generators up the finest scale while injecting noise in each scale. Both the generators and discriminators have the same receptive field, capturing structures of decreased size going up the pyramid field. At the crude scale, the generation is pure meaning spatial white Gaussian noise Zn is mapped onto image sample Xn.

$$Xn = Gn(Zn) \quad (1)$$

The receptive field being effective is at ~1/2 of the height of the image, thus Gn generating object's global structure and general layout. Each of generators at each scale (n<N) add details not generated on previous scales, with spatial noise zn, each generator Gn takes the up sampled image versions.

$$Xn = Gn(Zn, (Xn+1)\ ^r), n<N \quad (2)$$

The generators in the pyramid have similar architecture. The noise Zn is added to image (Xn+1)^r, being fed into the sequential convolutional layers. The GAN does not disregard the noise, as often that happens in random schemes conditionally. The convolutional layer's main role is generating missing details (Xn+1) ^r that is residual learning.

$$xn = (\tilde{}xn+1) \uparrow r + \psi n(zn + (\tilde{}xn+1) \uparrow r), \quad (3)$$

Gn operate the given function where ψn is fully convolutional net with 5 conv-blocks of Conv(3x3)-BatchNorm-LeakyReLU. 32 kernels per block are initiated at the

crudest scale and increase by the factor of 2 every 4 scales. By changing the dimensions of the noise maps, we can generate images of arbitrary size and aspect ratio at test time.

From the largest to the smallest scale, the multi-scale architecture is trained successively. Each GAN is fixed after its initial training. The reconstruction term and the adversarial term make up the nth GAN's training loss,

$$\min_{G_n} \max_{D_n} L_{adv}(G_n, D_n) + \alpha L_{rec}(G_n) \quad (4)$$

The adversarial loss, the distribution of patches in general samples xn and the distance between the patches in xn are penalized for Ladv. The collection of noise maps that generate samples xn are maintained by the Lrec (reconstruction loss).

3.2.2 Adversarial Loss

The generators Gn and Dn Markovian discriminator work together to determine whether each area of overlap is fake or real. In order to ensure training stability, we utilize the WGAN GP loss, where the overall final discrimination score is obtained by averaging over the patch discrimination map. Instead of employing random batch sizes, we calculate the loss across the entire image, enabling the network to learn boundary conditions, which is a crucial aspect of our setup.

3.2.3 Reconstruction Loss

To ensure a random noise input exist to general original image x,

We choose,

$$\{z_{rec\,N}, z_{rec\,N-1}, \ldots, z_{rec\,0}\} = \{z^*, 0, \ldots, 0\}, \quad (5)$$

where some fixed noise map is z*. (Kept fixed during training). When using these noise maps, denote by ˜xrec n the generated image at the nth scale.

Next for

$$n < N, \text{Lrec} = \text{Gn}(0,(\tilde{x}\text{rec } n+1) \uparrow r) - xn2, \text{ and for } n = N, \text{ we use Lrec} = \text{GN}(z*) - xN2 \quad (6)$$

The reconstructed image ˜xrec (5) n plays a training role to determine the standard deviation σn of the noise zn in each scale. Taking σn to be proportional to the root mean squared error (RMSE) between (˜xrec n+1) ↑r and xn, giving an indication of the number of details that need to be added at that scale.

We used the Google Colaboratory Notebook to make use of this GAN. With the help of the web scripting tool Google Colaboratory, we can employ the required GPU to work on programs that have higher running expenses. Since our GAN needs a GPU to function, Google Colaboratory was the perfect tool to use because it offered a 16 GB GPU Tesla T4 for the required training. We set up the requisite libraries and the latest version of PyTorch for the GPU call:

- **'Numpy'**: This library handles arrays and mathematical operations. It offers a reliable N-dimensional array object and a large selection of mathematical operations that relate to arrays.
- **'Matplotlib'**: Users can create static, animated, and interactive visualizations using the Python charting toolkit matplotlib.
- **'OpenCV':** It is a set of programming tools that are primarily made for real-time computer vision. It is extensively used in applications including object identification and recognition, image processing, video analysis, and others. It includes interfaces for C++, Python, and Java and supports a variety of operating systems, including Windows, Linux, and macOS. This library, which is often used

in computer vision and machine learning projects, provides a variety of tools and functions for working with images and videos.

- **‘Pillow’**: Users of Python can alter images using this open-source package. It includes a wide range of image processing features and supports many different images file types, including JPEG, PNG, and BMP. It can be used to edit images by rotating, resizing, and cropping them. Additionally, it aids with opening and editing photos. It is also used to create thumbnails, filter images, improve images, change modes, and convert file formats. It is an easy-to-use package that may be used for simple image processing applications. It has significant advantages for computer vision, machine learning, and artificial intelligence.
- **‘SciPy’**: It is a Python library for technical and scientific computing that is open-source and free. It provides a wide variety of mathematical tools and techniques for jobs like optimization, integration, interpolation, eigenvalue problems, etc. It is constructed over the NumPy library. It is frequently used in scientific research, engineering, physics, finance, and these fields. When combined with other scientific libraries like NumPy and Matplotlib, SciPy may be used to manipulate and visualize data even more effectively and robustly.
- **‘Imageio’**: An open-source library provides a straightforward and user-friendly interface for reading and writing a range of image and video formats. manipulation and visualization. To give even more powerful image modification capabilities, Imageio can be integrated with OpenCV and Pillow in addition to being built on a variety of other libraries, including NumPy and SciPy.
- **‘Scikit-image’**: Scikit-image is a popular Python image processing library (often referred to as skimage). It is an open-source toolkit including several algorithms for tasks like image restoration, segmentation, feature extraction, and more. It

was built on top of the SciPy and NumPy libraries. Scikit-image is widely used in machine learning, computer vision, and image processing. It is well-documented, highly strong, and has a large variety of purposes. It may be used with image processing tools like OpenCV and Pillow as well as machine learning tools like scikit-learn and TensorFlow.

The 4-channel image is then updated as the training image in the input directory, and the training bash command is then executed. Due to the precise configuration needed for accurate outcomes, training GAN is challenging. Since conditional probability is the foundation of GAN, it is challenging to train for the convergence necessary for excellent outcomes. It requires a change in the noise amplitude, loss function, or epochs. Although training the model takes a lot of time, we can train additional photos in the same model.

No. of training X-ray images	Approx. size of input image (4-channel)	No. of epochs (iterations)	Learning rate	Lambda gradient	Kernel size (K.S)	Additive noise continuous weight
14	650 x 275	2500	0.0005	0.25	3	0.09
Generator Steps	**Discriminator Steps**	**Noise continuous weight**	**Number of layers**	**Stride**	**Padding size**	**Reconstruction loss weight**
3	3	0.09	5	1	K.S/2	10

Table 4 Configuration of SinGAN-Seg model for optimum results.

As a result, artificial images will be created for the following phase.

3.3 Comparing real images with synthetic images

3.3.1 Fréchet Inception Distance

It is possible to compare our developed medical images to genuine datasets to train identification models since we have created medical images for further training. Fréchet Inception Distance (FID) score has been a crucial solution statistic since GANs were first introduced. A popular evaluation metric for generative models is the

FID score. By calculating the separation between their Inception activations, it compares the feature representations of generated images to those of actual images [53]. The score compares the two groups and gives the statistics on computer vision elements of the original photos that were obtained using the inception v3 model for image categorization.

The FID score is calculated as the following:

1. Compute the Inception activations for the real images and generated images.
2. Calculate the mean and covariance matrix of the activations for each dataset.
3. Calculate the Fréchet distance between the two Gaussian distributions defined by the activations' means and covariances.

The Fréchet distance, a metric for comparing the similarity of two probability distributions, is the separation between the distributions' means, with the separation between their covariances serving as a weight.

$$\mathbf{FID = ||mu\_r - mu\_g||^2 + Tr(Cov\_r + Cov\_g - 2(Cov\_r * Cov\_g)^{(1/2)}} \quad \mathbf{7}$$

Where mu_r and mu_g are the means of the real and generated data activations, Cov_r and Cov_g are the covariance matrices of the real and generated data activations, respectively, Tr is the trace of a matrix and ||.|| denotes the Frobenius norm. A lower number indicates that the generated images are more comparable to the genuine ones. The FID score runs from 0 to infinity.

Following steps are followed for evaluation of images using FID score:

1. Collect a dataset of real images.
2. Generate images using the generative model that you wish to evaluate.

3. Compute the activations of the Inception model for both the real images and the generated images.
4. Compute the mean and covariance matrix of the activations for each dataset.
5. Calculate the FID score using the means and covariances of the activations.

3.3.2 Inception Score

A popular evaluation metric for generative models that seeks to gauge the caliber and variety of generated images is the Inception Score (IS). It adheres to the principle that classifying high-quality images should result in labels with high entropy. Based on the class labels predicted by the inception model, it is based on the conditional likelihood of the generated images.

To calculate the IS score, the following steps can be followed:

1. Collect a dataset of real images.
2. Generated images using the generative model that you wish to evaluate.
3. Compute the class labels predicted by the inception model for both the real images and the generated images.
4. Compute the marginal probability distribution p(y) and the conditional probability distribution p(x|y) for both the real and generated images.
5. Calculate the Inception Score as the exponential of the average KL divergence between the conditional class labels distribution of the real images and the generated images:

$$IS = \exp(E_X[KL(p(y|x)||p(y))]) \quad 8$$

Where KL is the Kullback-Leibler divergence and E_x is the expectation over x.

It’s important to mention that IS score ranges between 1 and infinity with a high score indicating that the generated images are of higher quality and greater diversity and visual appeal.

Because multiple evaluation measures might capture distinct elements, it is vital and worthwhile to use two of these evaluation metrics.

3.4 U-Net models for classification and image segmentation

We introduce U-Net from Keras library and train 6 different variants of the models. This is not only to show which suits best for our image classification of the Charcot disease but also to show that Models can be trained on synthetic generated dataset to predict when an original image is used as sample.

Industries frequently use the U-Net architecture for image segmentation and classification applications.

The procedures below can be followed in general to train a U-Net model:

1. To begin the task at hand, gather a dataset of images and associated labels (segmentation or classification).
2. If necessary, preprocess the data (e.g., resizing, normalization).
3. Use a deep learning framework, such as Keras or TensorFlow, to define the U-Net architecture.
4. Set the loss function, optimizer, and any other training metrics before compiling the model.
5. Set the batch size, number of training epochs, and any other pertinent training parameters before training the model on the dataset.
6. Validate the model using a held-out validation set to keep an eye on overfitting and assess the model's effectiveness.

After training is finished, the model can be applied to new image predictions.

Hyperparameters are adjusted during training to get the best performance. These variables consist of the learning rate, batch size, and training epoch count. The following libraries are also necessary for the procedure:

- **‘OS’**: OS is the name of this Python module for interacting with operating systems. You can use the file system to perform operations like creating and removing folders.
- **‘Numpy’**: This library manages arrays and numerical operations. It provides a robust N-dimensional array object and a wide range of array-related mathematical operations.
- **‘Matplotlib.pyplot’**: The charting tool matplotlib.pyplot allows Python users to produce static, animated, and interactive visualizations.
- ‘**tensorflow.keras.optimizers. Adam’**: Adam is an optimization procedure that can be used to change the weights of a neural network. It is the go-to remedy for many deep learning problems.
- **‘datetime’**: You can change times and dates using Python's datetime module. Date and time objects are creatable and modifiable.
- **‘CV2’**: A free computer vision library. The collection contains around 2500 algorithms. These algorithms can be used to identify landscapes, detect related images in an image database, erase red eyes from flash photos, track eye movements, and create overlay markers. Additionally, they can be used to detect and recognize faces, identify objects, categorizes human actions in videos, track camera movements, track moving objects, extract 3D models of objects, create 3D point clouds from stereo images, stitch images together to create high-resolution images of entire scenes, and extract 3D models of objects.

- **‘PIL’**: The Python interpreter can manipulate images thanks to the Python Imaging Library (PIL). The library allows you to view, modify, and save a wide range of picture file formats.

In this instance, we'll use a variety of U-net types from the Keras-U-net-collection to verify the training of several U-net models on our dataset.

U-Net architecture types

These will include:

1) U-net with ImageNet trained RedNet15V2 backbone:

This deep learning U-net image segmentation technique modifies the architecture of the network by using a pre-trained RedNet15V2 as the encoder component. The RedNet15V2 model was trained on the ImageNet dataset. The features are taken out of the input by the RedNet15V2 model and sent to the decoder for additional processing. This pre-trained model gives U-Net robust characteristics to work with, enhancing model performance with the benefit of avoiding the need for model training from scratch and the associated time and processing resources.

2) U-Net Plus:

A modified version of the well-known U-net architecture adds more skip connections and an attention mechanism to alleviate some of the original architecture's drawbacks. The segmentation findings are preserved in high detail because to the attention mechanism, which enables the model to take only the most significant features and add additional skip connections to optimize information flow throughout the network. It offers improvement while keeping the implementation process simple.

3) Attention U-Net with ImageNet and VGG19 trained backbone:

This design includes an attention mechanism whose primary training source was an ImageNet-trained model. The segmentation outcomes are enhanced by the model's ability to selectively focus on key features thanks to the attention mechanism. This architecture offers the U-Net a robust collection of features to work with, enhancing the model's overall performance. It benefits activities that call for high-resolution feature maps.

4) U-Net with ImageNet trained EfficentB7 backbone Without Weights:

The mentioned model is the same as this one. The EfficientB7 model initializes in this instance without specified weights, though. With the remainder of the U-net architecture, it used random weights and fresh trains. As a result, this strategy requires more computational resources and time than using pre-trained weights, but it enables fine-tuning the EfficientB7 model for the task at hand.

5) Recurrent Residual (R2) U-Net:

Adding a recurrent residual block (R2 block) to the architecture will improve performance in this additional U-net variant for image segmentation jobs. To enhance outcomes, this block is made to capture long-range dependencies in the supplied data. For better network-wide information flow, the R2 U-net incorporates a second skip connection. This model not only helps with high-resolution feature map jobs, but it also makes the model more accurate.

6) Attention U-Net from scratch (no backbone or weights)

The same architecture as before is present here. Determining the architecture of this model and initializing random weights without any previously trained hyperparameters is the process of training it from scratch. This gives the model a

benefit when there is no pre-trained model available for the particular tasks or data area.

The model can be adjusted for greater optimum with more flexibility and adaptability thanks to this training from scratch, albeit it will cost more computational resources and require a larger dataset.

Pre-trained models can add bias from earlier trials or data that may not be ideal for the new tasks, even while they are good for initialization points.

Obs.	Model type	Learning rate	activation	Filter number	Batch Size	Epochs/iterations
1	U-Net	1e-3	LeakyReLU	(32,64,128,256,512)	4	50
2	U-Net Plus	1e-3	LeakyReLU	(32,64,128,256,512)	14	50
3	Attention U-Net	1e-3	LeakyReLU	(32,64,128,256,512)	5	50
4	U-Net	1e-3	LeakyReLU	(32,64128,256,512,1024)	10	50
5	Recurrent-Residual U-Net	1e-3	ReLU	(32,64,128,256,512,1024)	5	50
6	Attention U-Net	1e-3	LeakyReLU	(32,64,128,256,512,1024)	3	50

Table 5 shows the parameters adjusted for each model training.

3.5 Intersection over union (IOU) Score

Semantic segmentation is a technique frequently applied to medical imaging. What is in the image and where is it positioned inside the image are the two questions that are examined in relation to images in computer vision. To be more specific, dense prediction is the labelling of an image's pixels to each object (class) that is displayed. Labeling a group of pixels that constitute separate categories is thus crucial to deep learning systems. It is very helpful in a variety of identification applications. Machines

can supplement radiologists' analysis in our method of diagnosing medical images, saving time and money on test diagnoses.

Medical photos have been generated with masking so that problems in a particular case can be identified. For instance, detecting brain malignancies, bone fractures, etc. The IOU is a figure that expresses how much two boxes overlap. IOU assesses the Ground truth in the segmentation situation. It is the proportion between the overlap and combined prediction and ground truth areas. The value can be between 0 and 1, with 0 denoting no overlap at all and 1 denoting complete overlap. With this, we decide the prediction is **True Positive (TP), False Positive (FP),** or **False Negative (FN).** IOUs threshold is typically set at 0.5 but depends on the circumstance. The threshold 0.5 works well in our situation because the overall structure of the photos is similar. Considering this, comparing the real image to the ground truth images is a useful item detection measure.

4 RESULTS

In this chapter, we will present the outcomes and findings from our proposed methodology. This chapter serves as the core of thesis, where our research objective is addressed and evaluated with gathered evidence. By presenting in a concise and structured manner, this chapter highlights key discoveries, patterns and relationships observed in the investigation.

4.1 Data pre-processing

For creating synthetic images, we use the SinGAN-seg model for training via google Collaboratory. Google Collaboratory allows access to GPU that is available which is a requirement for the training. A 16 GB memory of Tesla T4 type is available for training on google Collaboratory that is the minimum requirement. First, we create a 4-channel image for the x-ray images being trained upon. Using ImageJ opensource tool, we crop out unnecessary image details to content and create the masks for further process. We also convert the images from JPEG format to PNG which enables better quality storage and the format of input for the training. We also resize from larger scale to a 0.3 times size which is an approximate of 650 x 250-pixel size. This is since we need an optimum image size for training to avoid over-fitting and reduce our time and resource costs. We then use the Gimp 2.0 software for merging the mask into the alpha channel. Our image consists of RGB 3 channels and the 4^{th} channel as the alpha channel that acts as the transparency layer.

4.2 GAN training and results

For SinGAN-seg process, the following dependencies are installed:

- Imageio version 2.21.1
- Numpy version 1.23.1

- Matplotlib version 3.5.3
- Scikit learn version 1.1.2
- SciPy version 1.9.0
- Tqdm version 4.64.0

Load the files onto the google Collaboratory notebook and install the dependencies. Upload the 4_channel image in the image input directory that would be called upon for training by the train.sh scripting.

GANs in general are difficult to train, thus hyperparameter are to be adjusted for convergence towards optimal results. For our GAN we adjust the hyperparameters as follows:

- Hyper parameters:

  Kernel size: default = 3

  Number of layers: default = 5

  Net padding size: default = 0

  Stride: default = 1
- Pyramid parameters:

  Pyramid scale factor: default = 0.75

  Noise_ amplitude (continuous noise weight): 0.09

  Image minimal size at coarser scale: 25

  Image maximal size at coarser scale: 250
- Optimization hyper parameters:

  Number of epochs to train per scale: 2500

  Schedular gamma: 0.1

  Learning rate: 0.0009

Generator inner steps: 3

Discriminator inner steps: 3

Lambda gradient (gradient penalty weight): 0.25

Reconstruction loss weight: default = 10

For the x-ray data, training a model over a 4-channel picture requires roughly 1 hour and 15 minutes. Depending on the input image's pixel size and the image itself, it would alter. It would be costly in terms of time and resources depending on the precise quality of the training image. For our experiment, the image size was chosen to prevent underfitting, overfitting, time, and resource costs.

The ability to learn another 4-channel image in the same notebook is a benefit of training the SinGAN-seg. indicating that the same training materials can be utilized repeatedly.

The SinGAN-seg creates random artificial images at various scales. The benefit of this GAN is its capacity to maintain the object's inherent universal structure unless requested. The amount of alteration that can be made in the photographs depends on the scale, from lowest to largest. For example, an image at scale 0 would be a complete reconstruction of itself but scaling up results in minimal modifications. We select a scale 1 image for our goal in the following model training since it generates changes that can be used as measurements.

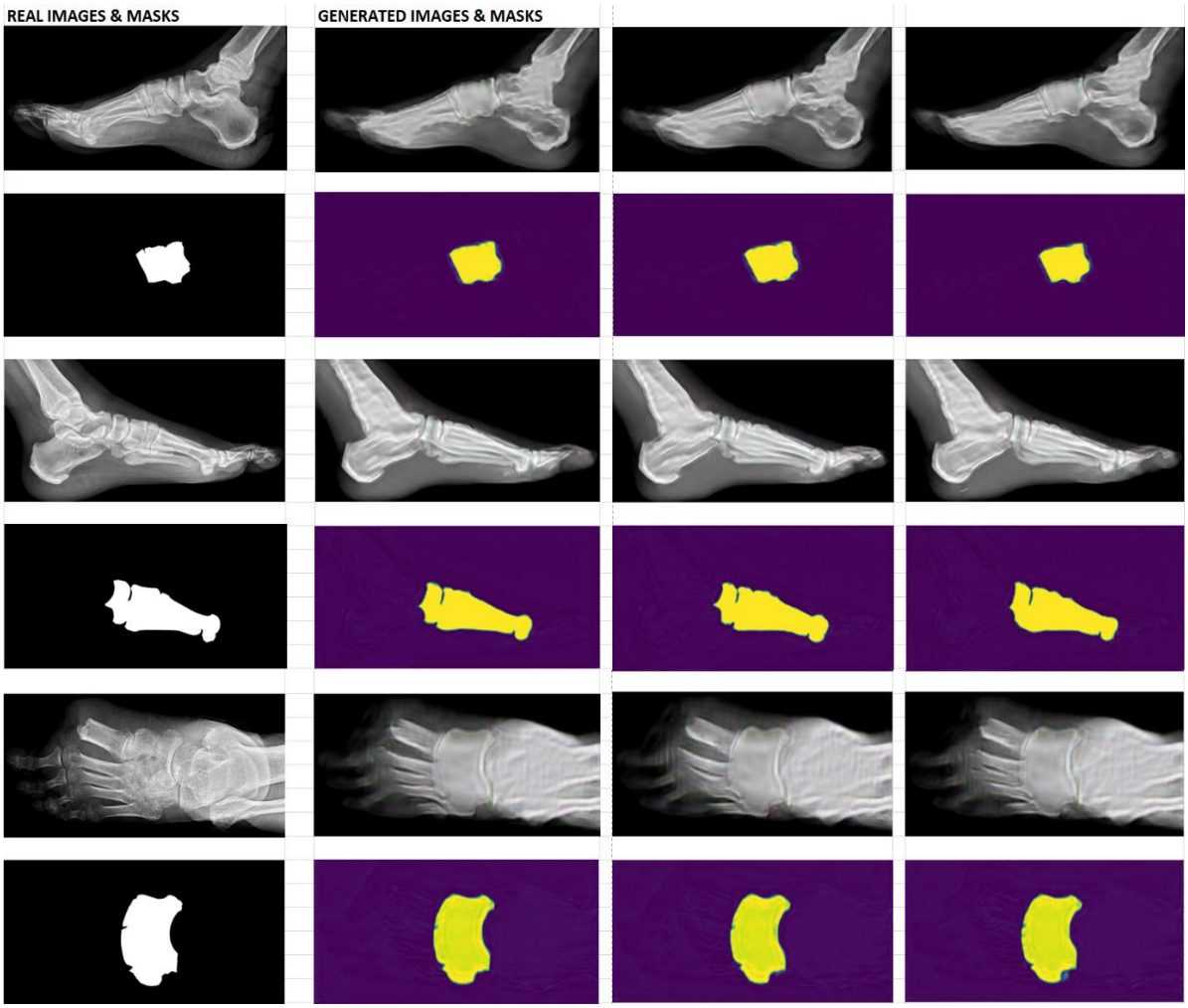


Figure 27 SinGAN-seg trained three different images and their respective masks input, creating three models, allowing to generate synthetic output while preserving the overall structure. The scale used allows for slight modifications to be made without compromising the universal structure of the image.

As shown in Figure 27. the synthetic images created are not up to mark with edging details. In our case of bone X-rays, the edge details of the bones need to be more defined as it is the most important feature for data collection for image segmentation. The style transferring algorithm was developed with that objective in mind. To transfer the style, there are various configurations that may be employed, and the default setting for the number of epochs is 1000. We use the algorithm to apply the style of the input image to the resulting synthetic ones. If the universal structure is the same, we can also use style pictures from another image to apply to the desired image. Thus Figure 28. shows the changes depicted after the style transfer has been

made. In this paper, we take the style from the input image and transfer it to the generated images. The style to content ratio 1: 1000 was taken and 16 GB tesla T4 type GPU was utilized for the process. It takes 18 seconds for the process to complete.

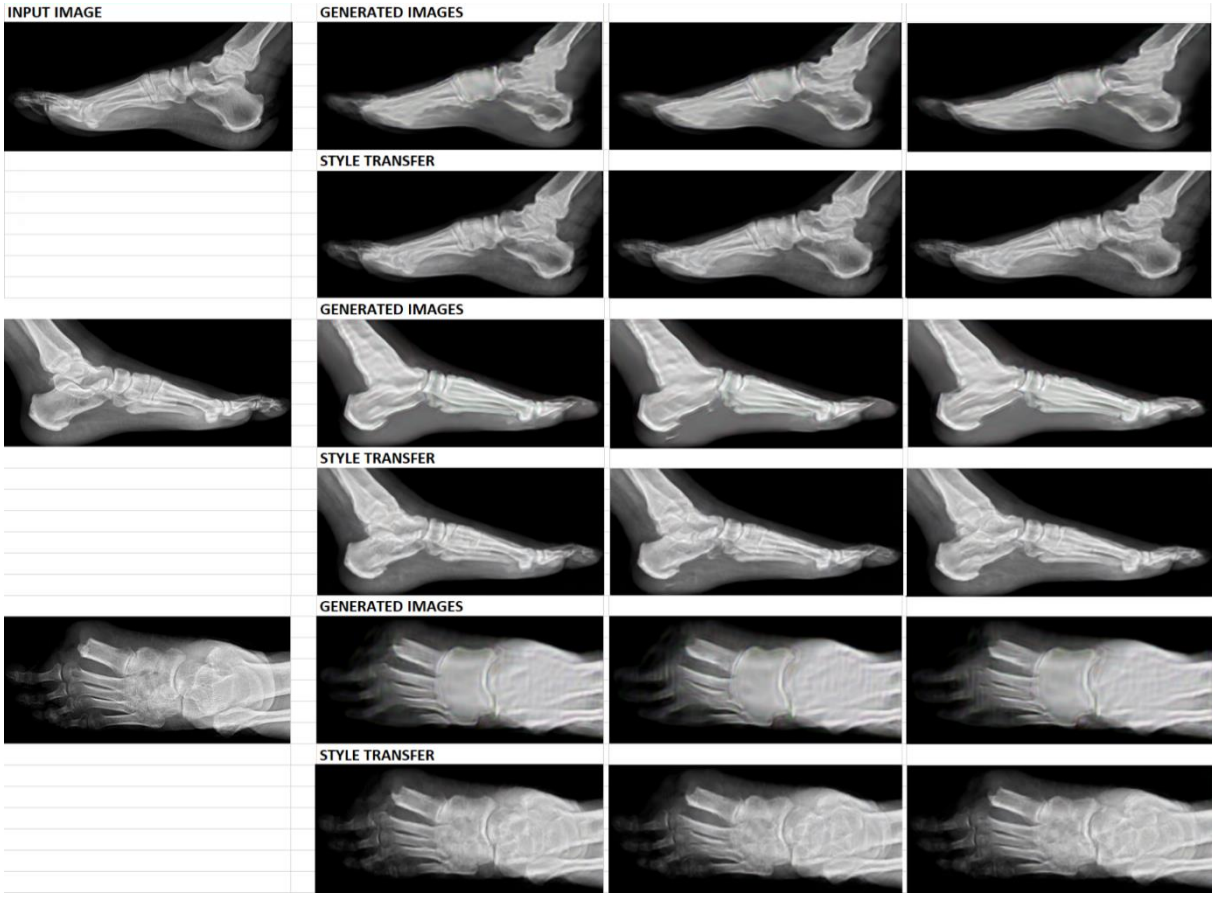


Figure 28 illustrates how style transfer can help develop images with additional details that are visible. Between synthetic images and their style-transferred images, there is a noticeable visual improvement. The second row with style transferring appears to provide higher quality.

4.3 FID score

A total of 14 original images were used to reconstruct 560 synthetic images. With visual representation, we also require metrics to assess the usefulness of synthetic images by contrasting them with the real images. We determine the separation between the feature vectors computed for actual and produced images using the Fréchet inception distance (FID). With a perfect score of 0.0 denoting identical, the

lower score will imply that the two groups of photos are more comparable and share similar data.

As the evaluation GAN-generated images often involves the use of this metric, we also employ the Inception score to make comparisons. The score is calculated in a way that higher values indicate better-quality images. For this purpose, we select two training images and calculate the FID score based on randomly chosen set of 10 generated images. Results were provided in table 6.

The generated images corresponding to their real training images are similar if their FID scores are low depending on the algorithm and conditions you have placed.

Generated Images A	**Original Image A**	**Generated Images B**	**Original Image B**
SIFID Score	**(As Reference)**	**SIFID Score**	**(As Reference)**
Image 1	112.624681	**Image 1**	196.9133563
Image 2	97.53735321	**Image 2**	153.0880456
Image 3	77.55707639	**Image 3**	187.1537887
Image 4	113.7532085	**Image 4**	161.3026366
Image 5	101.4250138	**Image 5**	171.6562017
Image 6	112.4381324	**Image 6**	156.5474843
Image 7	125.8133925	**Image 7**	160.2341075
Image 8	100.9401481	**Image 8**	149.0759948
Image 9	99.08005014	**Image 9**	152.4348914
Image 10	102.2032255	**Image 10**	139.9210889
Mean FID Score	104.3372281	**Mean FID Score**	162.8327595
Std. Deviation of FID Scores	12.24671672	**Std. Deviation of FID Scores**	16.72339666

Table 6 shows the Single image FID score (SIFID) in the range of 77 to 113 for training image A and 139 to 197 for training image B. The mean and std. deviation shows that, under our circumstances, generated images are structurally quite comparable to the original image.

The Fréchet Inception Distance (FID) score is not deterministic and can vary depending on several variables, including the model's architecture, the size of the training dataset, the caliber of the real images, and the task being assessed. The overall structure of the forms in the images is explicitly stated in our GAN design to be maintained, and as we specified no modification in it, the conditions were maintained in the generated images.

If the generated images and the genuine image look visually comparable, then the created images may have a similar structure and visual content. The FID score is a gauge of how closely the feature distributions in the generated and real images resemble one another. The score range for both images in Table 6. is under our condition optimum, which denotes a more comparable distribution of characteristics between the produced and real images.

4.4 Inception Score

We also take in the metric to calculate the Inception Score (IS) to ensure that the quality of the generated images is quite like the original image. Thus, we carried out experiments to find out the finding.

Obs.	Image A	IS Score	Obs.	Image B	IS
0	Image_A.png	(As Reference)	**0**	Image_B.png	(As Reference)
1	31_img_ST	0.001898	**1**	29_img_ST	0.001584
2	39_img_ST	0.001231	**2**	20_img_ST	0.001049
3	0_img_ST	0.001181	**3**	32_img_ST	0.001125
4	2_img_ST	0.001022	**4**	0_img_ST	0.001154
5	38_img_ST	0.001046	**5**	38_img_ST	0.001065
6	13_img_ST	0.001052	**6**	23_img_ST	0.001202
7	41_img_ST	0.001043	**7**	41_img_ST	0.001051
8	24_img_ST	0.001116	**8**	26_img_ST	0.001022
9	34_img_ST	0.001056	**9**	35_img_ST	0.001237
10	33_img_ST	0.001061	**10**	44_img_ST	0.001683
	Mean IS	0.0011706		**Mean IS**	0.001217
	Std. Deviation of IS	0.0002507		**Std. Deviation of IS**	0.000219

Table 7 displays tabulation for real images A and B with corresponding randomly produced images and IS ratings for comparison. The values of mean and std. deviation reveal that the generated photos are quite similar in quality to their training images.

We can see that minor differences show the images having similar quality and diversity. The plot graph in Figure 29 shows how much values difference exist between IS scores.

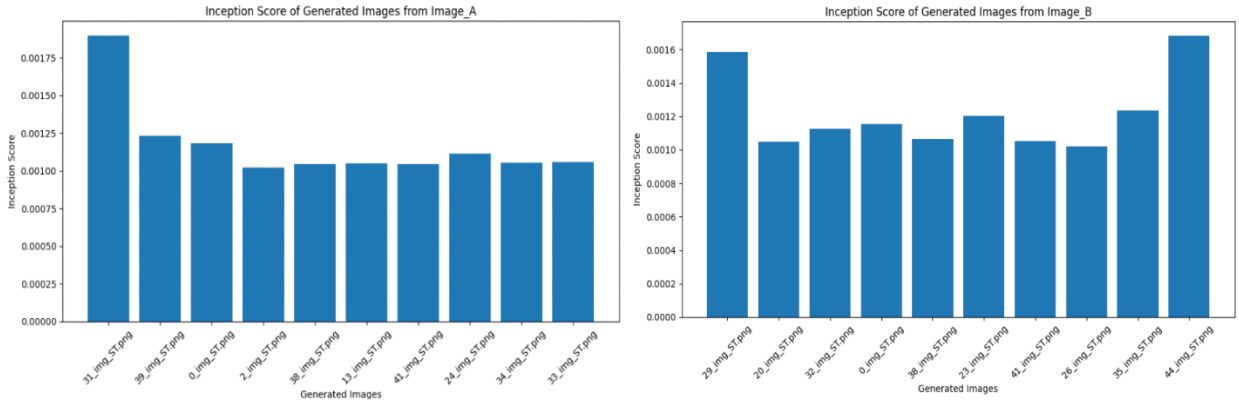


Figure 29 shows variations of IS scores of image A and image B with their corresponding generated images.

Both the FID score and the IS score exhibit value ranges with little variation, suggesting that the photos are very comparable. Consequently, it can be demonstrated that the resulting images are also CF X-ray images that have minor structure changes.

4.5 U-NET model training

Keras introduced the KERAS U-net collection through which 6 models of U-net variations were trained for semantic segmentation for the identification of CF through X-ray images.

Because of SinGAN-seg, we were able to produce a total of 560 X-ray images with corresponding 560 masks. These were taken as input and converted to .tiff format which is commonly used for medical data purposes. It doesn't reduce the quality of the images thus we move forward to create datasets of images and masks. We then resize the images to 128 and normalize. Using the train test split, we separate 90% of data for training and the other 10% for testing and view the images for Sanity check as shown in Figure 30.

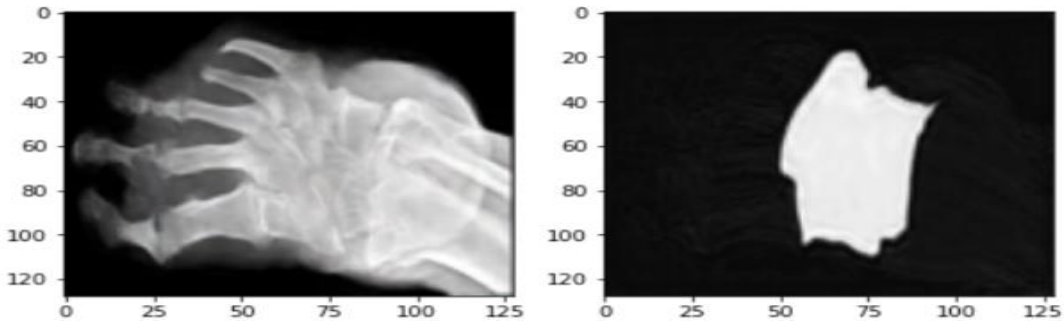


Figure 30 Training dataset normalized and resized for our U-Net models training.

For training the U-net models, we take the input shapes of the training images (image height, width, and channels) and move forward to our model training.

The training time for each model is as shown:

U-Net Model Type	Execution Time	Iterations completed
Model 1: U-net with ImageNet trained RedNet15V2 backbone	0:02:34.014152	24
Model 2: U-net Plus with ImageNet and VGG19 backbone	0:01:56.748635	31
Model 3: Attention U-net with ImageNet and VGG19 backbone	0:01:36.028577	17
Model 4: U-net without weights and EfficeintB7 backbone	0:10:19.398329	31
Model 5: Recurrent Residual (R2) U-net from scratch- no weights and backbone	0:07:43.450895	20
Model 6: Attention U-net from scratch – no weights and backbone	0:03:28.992021	19

Table 8 shows the execution time taken for each model to be trained with Early stopping with limitation of 50 epochs.

After multiple trials with each model with different backbones, we were able to train the best performed models of each. We collect and save the history of performance of models while training in csv format all the previous training data for each model.

The loss function, labeled as "binary cross entropy," is passed as the first argument supplied to the build() method. During training, this loss function is used to calculate

the error between the expected output and the actual output. In this case, it is a binary cross-entropy loss, which is commonly used for binary classification tasks.

The optimizer, which is stated as Adam(learning_rate = 1e-3), is the second argument passed. Adam is a particular kind of gradient descent technique that is typically used for deep learning model training. The default setting for Adam's optimizer is 1e-3 (0.001) for the learning rate. To reduce the loss function, this optimizer modifies the model's parameters.

Adam(learning_rate = 1e-3) refers to the optimizer, which is the second argument supplied. Adam is a specific type of gradient descent method that is frequently applied to the training of deep learning models. The learning rate for Adam's optimizer is set by default to 1e-3 (0.001). This optimizer changes the model's parameters to lower the loss function.

In conclusion, this code sets up all the six models to use accuracy and dice coef. as evaluation metrics, as well as binary cross-entropy loss, Adam optimizer, and a learning rate of 0.001. The fit() method can be used to train the model once it has been constructed.

The performance history data for each model, including loss and accuracy at each epoch, is then plotted.

Consequently, we plot the training and validation loss across the quantity of epochs. Both the training and validation loss values, which are taken from the history object, are set in the loss and Val loss variables, respectively. The number of epochs is created as an array using the range() method. The training loss is then plotted in yellow and the validation loss in red on the same graph using the plt.plot() method.

The legend is added to the plot using the plt.legend() function, and the x- and y-axes are labelled as "Epochs" and "Loss," respectively.

The training and validation dice coefficients are then similarly plotted over the number of epochs. Both the training and validation dice coefficient values, which are taken from the history object, are set in the acc and Val acc variables, respectively. The training dice coefficient is then plotted in yellow and the validation dice coefficient in red on the same graph using the plt.plot() method. The legend is added to the plot using the plt.legend() function, and the x- and y-axes are labelled as "Epochs" and "Dice," respectively.

The plot is finally shown on the screen using the plt.show() function. To summaries, we display the model's training and validation loss and dice coefficients across the number of epochs, which can be helpful to assess the model's performance and to spot potential overfitting or underfitting. While the training limit was set to 50 epochs, we also used the early stopping method that helped stopping from overfitting the models. Early stopping involves monitoring the performance of the model during training and stopping the training process early if certain criteria are met. The benefits of using Early stopping are:

- Prevents over-fitting as U-Net is prone to this when dealing with limited training data by monitoring on validation set and stopping the training process when performance starts to degrade, benefiting in generalizing well over data.
- It saves valuable computational resources and training time as stops the process once model achieves satisfactory rather than waiting for predefined number of epochs to complete.

- Improves model generalization over unseen data by preventing models from becoming overly complex and memorize noise in training data.
- Simplifies model selection by providing useful criterion for model selection by comparing performance of different models on validation set.
- Increases stability and reproducibility as final model will be consistent across different training runs while being stable.

The following are the plotted results of each of the models:

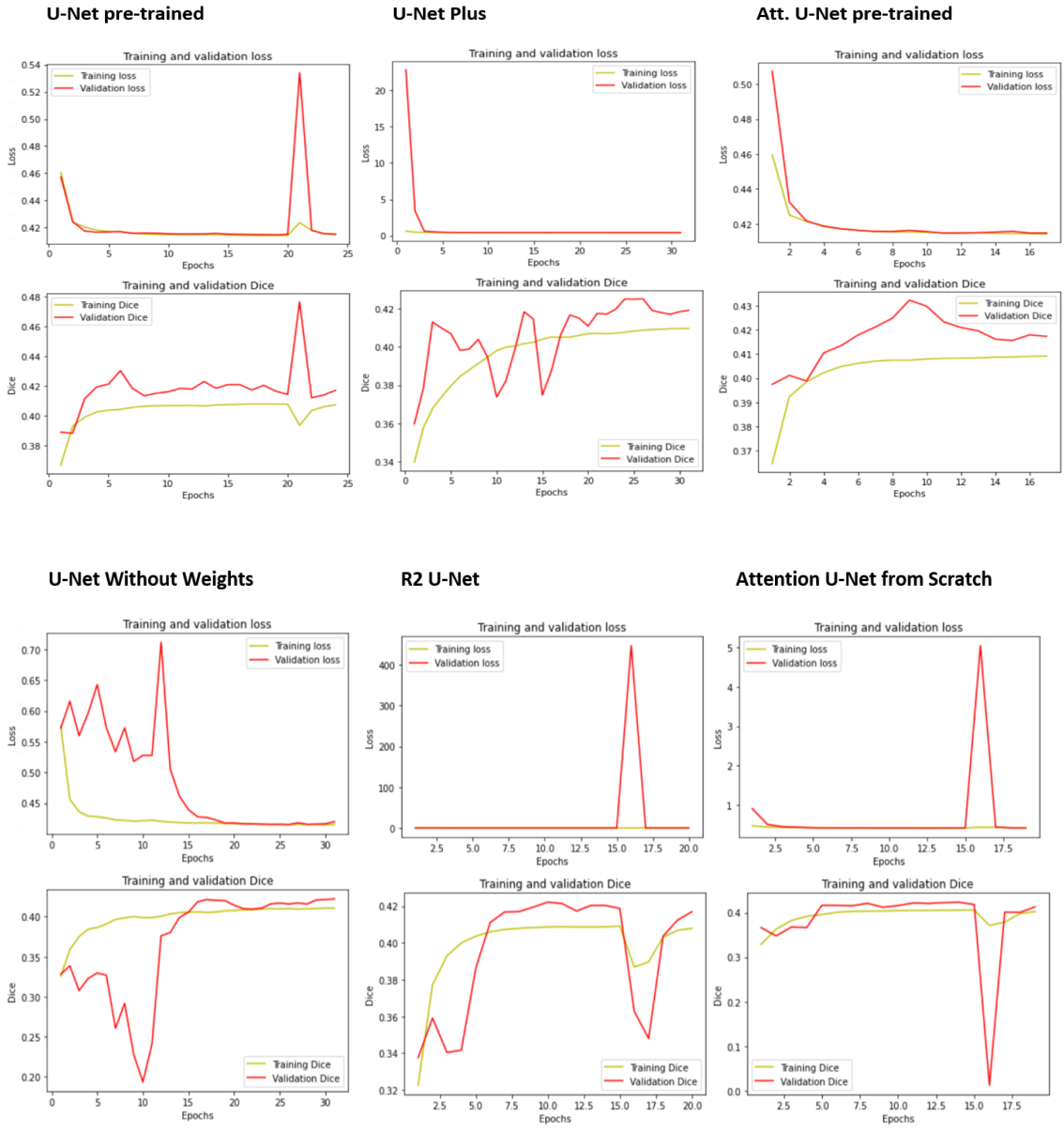


Figure 31 The graphs of loss and Dice of each model show the changes while training and validation of each model.

Understanding how well the model can match the training data during training is possible with the help of the loss plot. The model can match the training data more accurately the lower the loss. A model that overfits the training set of data has a low training loss but a large validation loss.

A statistic for assessing the efficacy of image segmentation methods is the Dice coefficient. It gauges how closely the expected and actual segmentations resemble one other. Perfect overlap is indicated by a Dice coefficient of 1, while no overlap is shown by a Dice coefficient of 0. A model is likely to be a good model if both the training Dice coefficient and the validation Dice coefficient steadily rise as the number of training epochs rises. Inferring from this that the model can generalize effectively and isn't overfitting the training set of data.

The optimal model will often be the one with the highest validation dice coefficient and lowest validation loss. With these ideas in mind, a comparison between the models can be made to determine the best and worst performance of the trained model.

Comparing the loss graph to check how much model was able to fit the training data, all the models showed a trend of low training loss and low validation loss. For model 1,5 and 6, the performance was quite good with low validation loss until the last epochs where a huge loss occurred in validation and was recovered in half an epoch later. Models 2 and 3 were quite stable in training with both training and validation loss being exponentially low. In the case of model 4, the initial epochs were difficult as validation loss fluctuations existed until later the model fit itself to perform with lower loss. This indicates that all the model performances were good. However, the difference of their performance was made visible by the dice coefficient metric plot

graph. A high training dice and low validation dice can signify overfitting, which means both the models perform well on the training data but struggle to generalize to new data. Due to its poor performance on unobserved data, the model in this instance is not regarded as a good model. Thus, would require regularization or early stopping to prevent overfitting.

4.6 Performance comparison

Model Type	Weights	Backbone	Performance trends
Model 1: U-Net	ImageNet	ResNet15V2	Demonstrated a noticeable trend of improved training and validation accuracy, which increased as the number of epochs was increased, indicating a favorable gradient.
Model 2: U-Net Plus	ImageNet	VGG19	The validation dice gradient fluctuated alongside the training dice gradient in the middle of epoch execution but improved later.
Model 3: Attention U-Net	ImageNet	VGG19	The model performed well by maintaining a higher validation dice score than the training dice score during training, indicating proficiency in predicting training data and generalizing to unseen validation data.
Model 4: U-Net	From Scratch	EfficeintB7	The model initially struggled with validation but improved later, suggesting poor generalization to unseen data. Overfitting to the training data might have occurred due to model complexity or limited training data.
Model 5: Recurrent Residual (R2) U-Net	From Scratch	From Scratch	Model 5 performs effectively with higher validation dice scores than training dice scores, indicating successful training with some variability. The model shows good generalization despite complexity and potential lack of overfitting. Overall, the model performs well despite unpredictable validation data, indicating successful training and generalization.
Model 6: Attention U-Net	From Scratch	From Scratch	The model experienced overfitting indicated by a drop in validation dice score after a certain point in training despite increasing stability. Although the model recovered, this indicates a need for improved generalization to unseen data.

Table 9 Each model is trained with specific weights and backbones showing the performance trends on the same training generated dataset.

Several pre-trained weights and backbone alternatives are available in U-Net, a common architecture for image segmentation applications, each with their own advantages that we have utilized from model 1 to 4 as shown in Table 9. Here is a brief list of their benefits:

Pre-trained Weights: U-Net models with pre-trained weights were developed using huge datasets and frequently a distinct task, such as image classification. As a starting point for transfer learning, these pre-trained weights enable quicker convergence and increased performance when fine-tuning on segmentation tasks. They record broader characteristics and patterns, which helps the model learn more quickly.

Example used in our training models in Table 9 is ImageNet. Millions of labelled photos make up the massive image database known as ImageNet, which is frequently used to train and test deep learning models for image classification tasks. It has been crucial in improving the field of computer vision research, stimulating algorithmic innovations, and acting as a benchmark dataset for assessing model efficacy. The advancement of deep learning methodologies and the development of computer vision have both been heavily influenced by ImageNet.

Backbones: The foundational architecture upon which the U-Net model is based is referred to as the U-Net backbone. Various backbone topologies, such VGG, ResNet, or EfficientNet, provide a variety of advantages:

- **ResNet**: With their skip connections, ResNet backbones solve the vanishing gradient issue and allow for deeper networks. They increase data flow and aid in capturing more complicated characteristics, which enhances segmentation performance.
- **VGG:** VGG backbones are renowned for being both straightforward and efficient. Due to their applicability for smaller datasets, they offer high feature extraction capabilities and are frequently utilized in U-Net models.

- **EfficientNet:** To obtain the best trade-offs between model size and accuracy, EfficientNet backbones use a compound scaling technique. They are excellent for contexts with limited resources because they use memory and computing resources so efficiently.

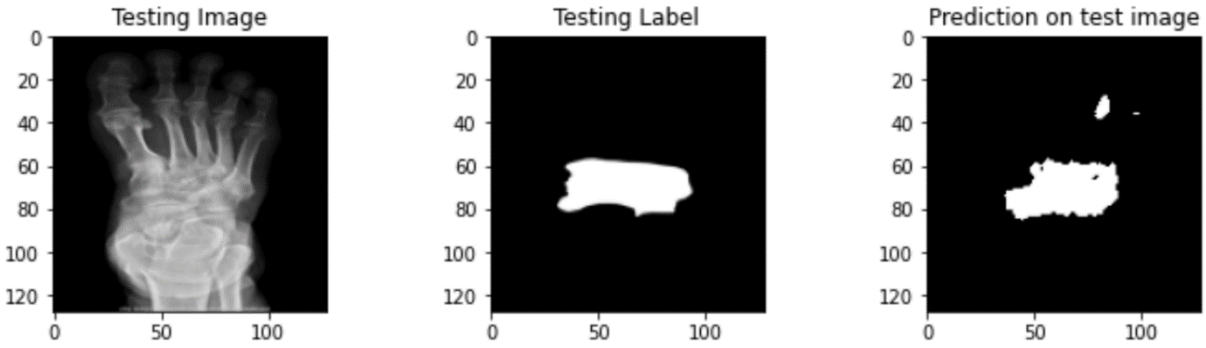
4.7 Visual model Prediction tests

The performance of the trained segmentation models is then assessed for prediction on test data. The tf.keras.models.load model() function is used to load the model initially. The next step is to choose a random image from the test dataset and to get the ground truth label for it. The loaded model is then applied to the chosen test image and its label to produce a prediction. The next step is to choose a random image from the test dataset and to get the ground truth label for it. The loaded model is then applied to the chosen test image and its label to produce a prediction. The plt.imshow() function is then used to plot and display the prediction, test image, and ground truth label.

We then determine the IOU for this solitary test image between the ground truth label and the prediction of 6 images that were not trained for synthetic images. The IOU is a metric that assesses the degree of agreement between expected and actual labelling; a value of 1 denotes a perfect match (that is theoretical), while a value of 0 denotes complete overlap.

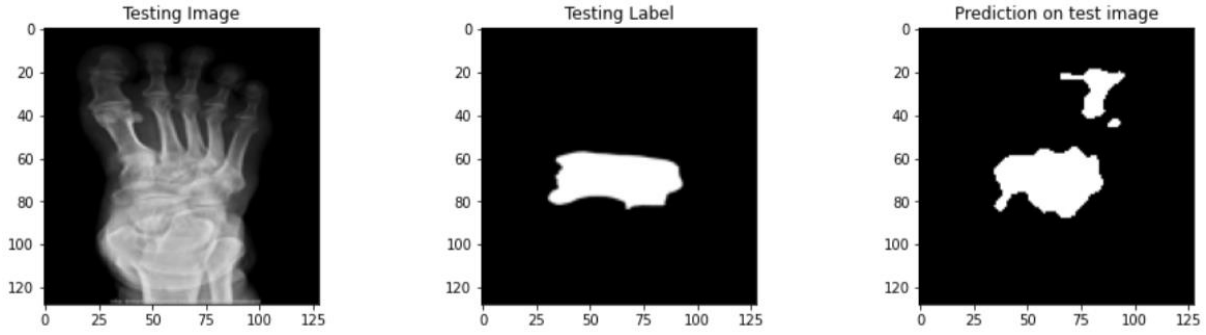
These are the displays of sample test images, test masks and the predicted masks of each of the models trained with their IOUs:

Model 1: U-Net



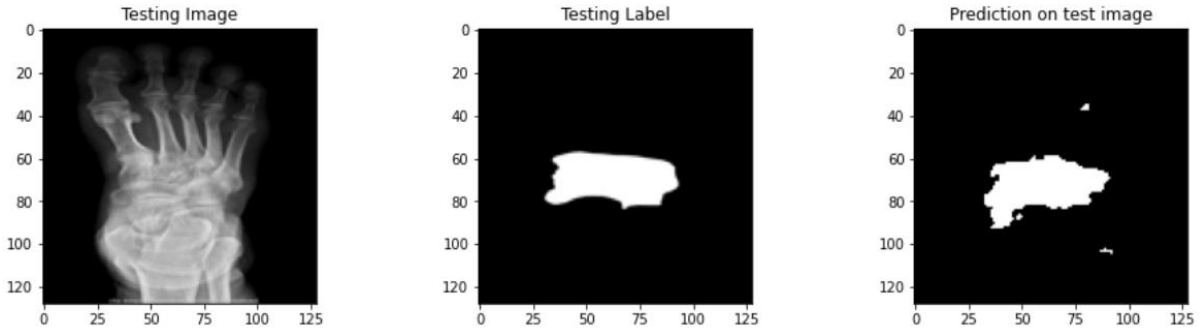
Single Image IOU	Average IOU of Dataset
0.8034092	0.71790123

Model 2: U-Net Plus



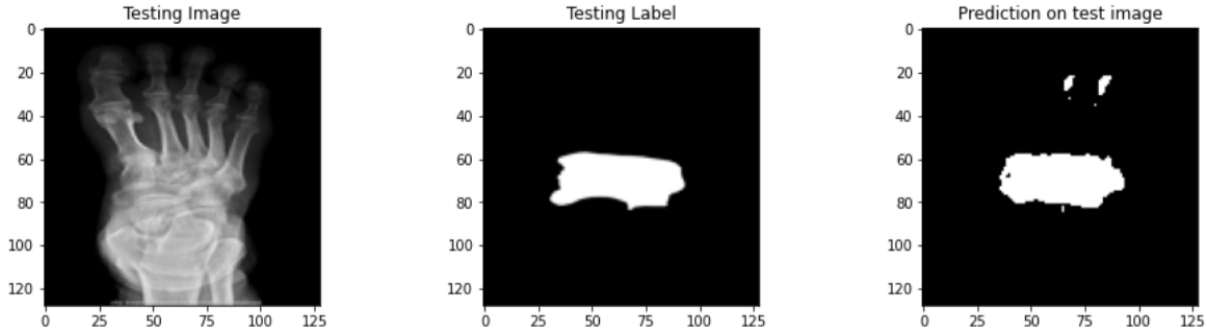
Single Image IOU	Average IOU of Dataset
0.7472584	0.64247054

Model 3: Attention U-Net with ImageNet trained backbone



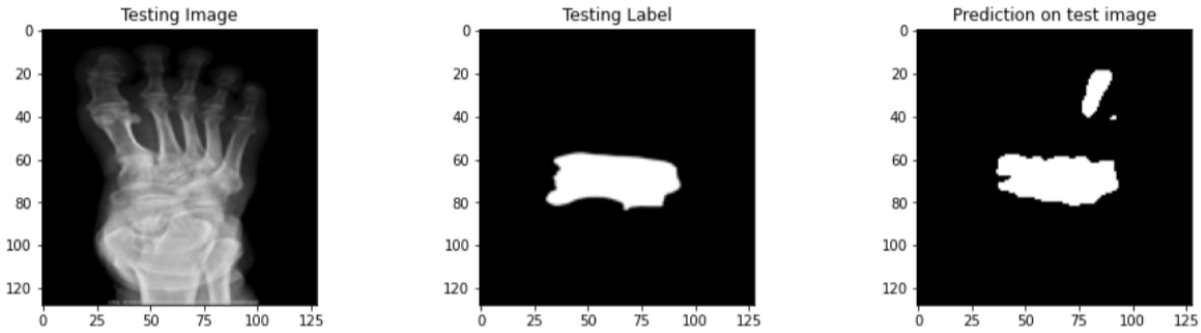
Single Image IOU	Average IOU of Dataset
0.812018	0.7275119

Model 4: U-Net with ImageNet trained without Loading Weights



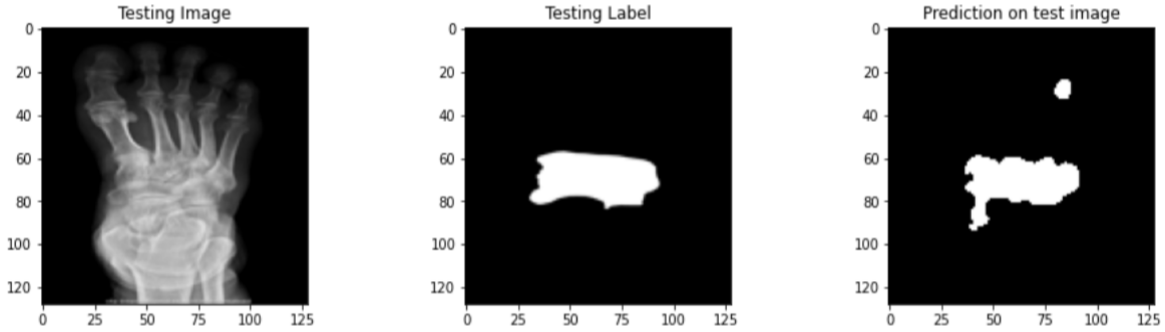
Single Image IOU	Average IOU of Dataset
0.88514024	0.71767753

Model 5: Recurrent Residual (R2) U-Net



Single Image IOU	Average IOU of Dataset
0.84098256	0.7571535

Model 6: Attention U-Net from Scratch – no backbone or weights



Single Image IOU	Average IOU of Dataset
0.8459928	0.7044198

Figure 32 Visualization of model predictions on a sample test CF x-ray image. An example single image IOU score for each model is provided, along with a comparison to the total average IOU score. When the IOU threshold is set at 0.5, the models produce very promising findings.

4.8 IOU Score

We also use the panda's library to generate the IOU values for each test image. We calculate the IOU of the images tested for visual prediction of each model and then it the mean average IOU of the sample test images and print it. The IOU between the prediction and ground truth label for each image is then determined using code that is implemented to loop through the whole test dataset. The average IOU for all test photos is printed after that as shown in Figure 32.

5 CONCLUSION & DISCUSSION

It is essential that AI and machine learning be applied in the healthcare system due to the rapidly growing use of these techniques in research. With the help of artificially generated datasets, we have shown in this study that training models for medical professionals to utilize in recognizing CF can be beneficial for improving segmentation accuracy and predictability. It is proven by the FID and Inception Score results that synthetic images can be compared to and share characteristics with the original X-ray images of the foot of the criminal. As a result, GANs can be applied to any image collection, specifically to increase the amount of data for use in medical research. As the synthetic data doesn’t belong to any living person, there won't be any ethical issues about patient privacy breaches. The ability to produce various image data with behaviors that can be applied to instructional purposes is another benefit of this.

Additionally, since the IOU scores on the test data were higher than the cutoff of 0.5, the u-net models are being used in industries to create quite accurate image segmentation models. However, we may improve the performance of the models even more by adding more data for training, which is exactly why generating synthetic images would prove useful where medical data collection is difficult due multiple reasons including ethical, privacy, rarity of disease, expensive tests etc.

Also, by experimenting with 6 different types of U-Net models, model 5: Recurrent Residual (R2) U-net turned out to be performing quite good with the given conditions. Since it was trained from scratch, the model was trained quite good against the other model under the same conditions of resources and epochs. It is a good sign that the model can generalize to new data when the validation dice consistently performs

better than the training dice. A further indication that the model is learning and developing throughout the training procedure is the constant growth in the training dice. The model is still functioning well overall, despite the validation dice fluctuating, which raises the possibility that the validation set contains some unpredictability. It's also important to note that the training loss is dropping slowly throughout the course of training, which shows that the model is improving and making fewer mistakes on the training set. A strong sign that the model is functioning well is the validation loss is also low and decreasing. It shows that the model is capable of accurately segmenting the validation data while also doing it with a minimal amount of error.

Comparative analysis

There are various things that can be done to better enhance the Recurrent Residual R2 U-Net model that can also be implied on the other tested models:

- Increasing the dataset size: The model should be able to learn more general features that will help it perform better with more data.
- Augmenting the dataset: The model will be exposed to more variations of the same images and be better able to generalize by randomly performing transformations to the images in the dataset, such as rotation, scaling, and flipping.
- The architecture of the model is being fine-tuned: To increase performance, the U-Net model's architecture can be changed to incorporate additional layers or layers of various kinds.
- By increasing computational resources, i.e., GPU would also allow us to increase filter numbers and numbers of epochs trained. This will not only allow us to train models with better performance but also reduce the time costs.

The model is designed to work for 2D images because the image segmentation is univariate. This model can be used as a decision-supporting tool by doctors to identify Charcot's disease in the early stages. This test can be useful and carried out under initial symptoms that would indicate CF disease exists. These include ulcerification, redness, pain, numbness in the foot.

Singan-seg when tested in its initial research of its performance with other GANs was able to produce the best results, indicating that they can produce synthetic images via just one image at best quality. However, since it utilized scale 0 under its own observations, we tested our images that were trained on scale 1 to keep the generalized overall shape of our x-ray images foot intact. Even though the results of synthetic image production were good, we still had limitations under that same general structure i.e., if the x-ray image trained is of a 50-year-old person, it will remain a 50-year-old persons synthetic images with internal changes. Thus, it can only be proposed to be comparative to local data i.e., national rather than international where average feet size varies.

This experiment still can be improved as many difficulties were raised during the process. In the case of training data, the original data is tough itself as a radiologist must make sure that the x-ray images provided shows clarity. Furthermore, rather more complex training data will be difficult to train and make predictions. In our case, not only the bone features were different, but the global structure pictorial angular view also varied. For future perspective, experiments can be made on the criteria that one of these features of the image’s dataset must be the same. Yet, a lot of future experiments can be conducted that could replace the doctor’s opinion of results. However, yet until AI and machine learning algorithm advance, this model is an aiding tool for doctors to determine the disease conditions. We still can conclude

a fact that there exists a future where AI will be integrated in medical imaging machinery such that it would be able to give us direct results of identification.

In the modern world, machine learning and AI technology have advanced significantly. It is imperative to employ these technologies in healthcare to learn how fantastic health may be sustained by incorporating technology. Computerization has expanded the digital world to aid the public in numerous activities.

6 REFERENCES

[1] A. B. Rosskopf, C. Loupatatzis, C. W. A. Pfirrmann, T. Böni, and M. C. Berli, "The Charcot foot: a pictorial review," *Insights into Imaging*, vol. 10, no. 1. Springer Verlag, Dec. 01, 2019. doi: 10.1186/s13244-019-0768-9.

[2] F. B. Ergen, S. E. Sanverdi, and A. Oznur, "Charcot foot in diabetes and an update on imaging," *Diabetic Foot and Ankle*, vol. 4. Nov. 20, 2013. doi: 10.3402/dfa.v4i0.21884.

[3] B. Bin Younis, A. Shahid, R. Arshad, S. Khurshid, and J. Masood, "Charcot osteoarthropathy in type 2 diabetes persons presenting to specialist diabetes clinic at a tertiary care hospital," *BMC Endocr Disord*, vol. 15, no. 1, Jun. 2015, doi: 10.1186/s12902-015-0023-4.

[4] "With 1 in 4 adults living with diabetes, Pakistan has highest diabetes prevalence in the world." [Online]. Available: www.diabetesatlas.org.

[5] M. Braiki, M. A. Khalifa, B. Faidi, M. Ghannouchi, and F. Derbel, "Introductory Chapter: Diabetic Foot." [Online]. Available: www.intechopen.com

[6] I. Ali Shaikh, N. Masood Sddiqui, and J. Hameed Shaikh, "Diabetic Foot Ulcer: An Easy and Comprehensive Approach." [Online]. Available: www.intechopen.com

[7] L. C. Rogers *et al.*, "The Charcot foot in diabetes," *Diabetes Care*, vol. 34, no. 9. pp. 2123–2129, Sep. 2011. doi: 10.2337/dc11-0844.

[8] T. Jin and M. Skalski, "Sanders CT classification of calcaneal fracture," in *Radiopaedia.org*, Radiopaedia.org, 2014. doi: 10.53347/rID-27025.

[9] S. Rajbhandari, R. Jenkins, D. C., and S. Tesfaye, "Charcot neuroarthopathy in diabetes mellitus," *Diabetologia*, vol. 45, pp. 1085–1096, Feb. 2002, doi: 10.1007/s00125-002-0885-7.

[10] M. M. and R. S. Ivo Schoots, "Osteomyelitis versus Charcot," *Radiology department of the AMC in Amsterdam and the Rijnland hospital in Leiderdorp, the Netherlands*, Mar. 06, 2011.

[11] L. SANDERS, "Diabetic neuropathic osteoarthropathy : The Charcot foot," *The high risk foot in diabettes mellitus*, 1991, Accessed: Oct. 11, 2022. [Online]. Available: https://cir.nii.ac.jp/crid/1571135650126367744.bib?lang=en

[12] C. Holmes, B. Schmidt, M. Munson, and J. S. Wrobel, "Charcot stage 0: A review and consideratons for making the correct diagnosis early," *Clin Diabetes Endocrinol*, vol. 1, no. 1, Dec. 2015, doi: 10.1186/s40842-015-0018-0.

[13] B. Najafi *et al.*, "Plantar temperature response to walking in diabetes with and without acute Charcot: The Charcot activity response test," *J Aging Res*, vol. 2012, 2012, doi: 10.1155/2012/140968.

[14] C. Whelan, "Charcot Arthropathy, Charcot Joint, or Charcot Foot," 2018. [Online]. Available: https://www.healthline.com/health/charcot-foot

[15] "Nearly 50 percent of patients with neuropa-thy had an associated plantar ulcer. 4." [Online]. Available: www.aafp.org/afp

[16] T. Ebeling and T. Sane, "TONI-KARRI PAKARINEN The Management and Clinical Outcome of the Charcot Foot," 2012.

[17] “The consequences of complacency_ managing the effects of unrecognized Charcot feet _ Enhanced Reader”.

[18] M. W. Sohn, T. A. Lee, R. M. Stuck, R. G. Frykberg, and E. Budiman-Mak, “Mortality risk of charcot arthropathy compared with that of diabetic foot ulcer and diabetes alone,” *Diabetes Care*, vol. 32, no. 5, pp. 816–821, May 2009, doi: 10.2337/dc08-1695.

[19] F. Jiang *et al.*, “Artificial intelligence in healthcare: Past, present and future,” *Stroke and Vascular Neurology*, vol. 2, no. 4. BMJ Publishing Group, pp. 230–243, Dec. 01, 2017. doi: 10.1136/svn-2017-000101.

[20] D. L. Pham, C. Xu, and J. L. Prince, “CURRENT METHODS IN MEDICAL IMAGE SEGMENTATION 1,” 2000. [Online]. Available: www.annualreviews.org

[21] “THE PERSONAL DATA PROTECTION BILL, 2018 CHAPTER I PRELIMINARY.”

[22] “PERSONAL INFORMATION PROTECTION ACT CHAPTER I. GENERAL PROVISIONS,” 2011.

[23] F. Renard, S. Guedria, N. De Palma, and N. Vuillerme, “Variability and reproducibility in deep learning for medical image segmentation,” *Sci Rep*, vol. 10, no. 1, Dec. 2020, doi: 10.1038/s41598-020-69920-0.

[24] M. J. Willemink *et al.*, “Preparing medical imaging data for machine learning,” *Radiology*, vol. 295, no. 1. Radiological Society of North America Inc., pp. 4–15, 2020. doi: 10.1148/radiol.2020192224.

[25] S. Yu *et al.*, “Robustness study of noisy annotation in deep learning based medical image segmentation,” *Phys Med Biol*, vol. 65, no. 17, Sep. 2020, doi: 10.1088/1361-6560/ab99e5.

[26] “Differential privacy in health research A scoping review”.

[27] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, “Generative Adversarial Networks: An Overview,” *IEEE Signal Processing Magazine*, vol. 35, no. 1. Institute of Electrical and Electronics Engineers Inc., pp. 53–65, Jan. 01, 2018. doi: 10.1109/MSP.2017.2765202.

[28] “Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks”.

[29] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-Attention Generative Adversarial Networks.” [Online]. Available: https://github.com/

[30] H. Alqahtani, M. Kavakli-Thorne, and G. Kumar, “Applications of Generative Adversarial Networks (GANs): An Updated Review,” *Archives of Computational Methods in Engineering*, vol. 28, no. 2, pp. 525–552, Mar. 2021, doi: 10.1007/s11831-019-09388-y.

[31] K. Kazuhiro *et al.*, “Generative Adversarial Networks for the Creation of Realistic Artificial Brain Magnetic Resonance Images,” *Tomography*, vol. 4, no. 4, pp. 159–163, Dec. 2018, doi: 10.18383/j.tom.2018.00042.

[32] V. Thambawita *et al.*, “SinGAN-Seg: Synthetic training data generation for medical image segmentation,” *PLoS One*, vol. 17, no. 5 May, May 2022, doi: 10.1371/journal.pone.0267976.

[33] T. Rott, S. Technion, T. Dekel, G. Research, and T. Michaeli, “SinGAN: Learning a Generative Model from a Single Natural Image.”

[34] L. A. Gatys, A. S. Ecker, and M. Bethge, “A Neural Algorithm of Artistic Style,” Aug. 2015, [Online]. Available: http://arxiv.org/abs/1508.06576

[35] R. Padilla, W. L. Passos, T. L. B. Dias, S. L. Netto, and E. A. B. Da Silva, “A comparative analysis of object detection metrics with a companion open-source toolkit,” *Electronics (Switzerland)*, vol. 10, no. 3, pp. 1–28, Feb. 2021, doi: 10.3390/electronics10030279.

[36] G. Litjens *et al.*, “A Survey on Deep Learning in Medical Image Analysis,” Feb. 2017, doi: 10.1016/j.media.2017.07.005.

[37] O. O’Driscoll *et al.*, “Object detection to compute performance metrics for skill assessment in central venous catheterization,” SPIE-Intl Soc Optical Eng, Feb. 2021, p. 39. doi: 10.1117/12.2581889.

[38] D. Müller, I. Soto-Rey, and F. Kramer, “Towards a guideline for evaluation metrics in medical image segmentation,” *BMC Research Notes*, vol. 15, no. 1. BioMed Central Ltd, Dec. 01, 2022. doi: 10.1186/s13104-022-06096-y.

[39] F. J. J. Joseph, S. Nonsiri, and A. Monsakul, “Keras and TensorFlow: A Hands-On Experience,” in *EAI/Springer Innovations in Communication and Computing*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 85–111. doi: 10.1007/978-3-030-66519-7_4.

[40] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017, doi: 10.1145/3065386.

[41] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” May 2015, [Online]. Available: http://arxiv.org/abs/1505.04597

[42] J. Long, E. Shelhamer, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” Nov. 2014, [Online]. Available: http://arxiv.org/abs/1411.4038

[43] S. Chen *et al.*, “U-Net Plus: Deep Semantic Segmentation for Esophagus and Esophageal Cancer in Computed Tomography Images,” *IEEE Access*, vol. 7, pp. 82867–82877, 2019, doi: 10.1109/ACCESS.2019.2923760.

[44] G. Gaál, B. Maga, and A. Lukács, “Attention U-Net Based Adversarial Architectures for Chest X-ray Lung Segmentation,” Mar. 2020, [Online]. Available: http://arxiv.org/abs/2003.10304

[45] P. Anderson *et al.*, “Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering,” Jul. 2017, [Online]. Available: http://arxiv.org/abs/1707.07998

[46] O. Oktay *et al.*, “Attention U-Net: Learning Where to Look for the Pancreas,” Apr. 2018, [Online]. Available: http://arxiv.org/abs/1804.03999

[47] “Recurrent Residual Convolutional Neural Network based on UNet”.

[48] A. N. Gajjar and J. Jethva, “Intersection over Union based analysis of Image detection/segmentation using CNN model,” in *2022 Second International Conference on Power, Control and Computing Technologies (ICPC2T)*, 2022, pp. 1–6. doi: 10.1109/ICPC2T53885.2022.9776896.

[49] D. López-López *et al.*, “Prevalence of foot ulcers in diabetic patients in Punjab, Pakistan.”

[50] J. Sheffield, “ImageJ, A Useful Tool for Biological Image Processing and Analysis,” *Microscopy and Microanalysis*, vol. 13, no. S02, Aug. 2007, doi: 10.1017/s1431927607076611.

[51] The GIMP Development Team, “GIMP.” Accessed: Jun. 12, 2019. [Online]. Available: https://www.gimp.org

[52] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, “UNet++: A Nested U-Net Architecture for Medical Image Segmentation,” Jul. 2018, [Online]. Available: http://arxiv.org/abs/1807.10165

[53] Y. Yu, W. Zhang, and Y. Deng, “Frechet Inception Distance (FID) for Evaluating GANs.” [Online]. Available: https://machinelearningmastery.com/how-to-implement-the-frechet-