ADVERSARIAL ATTACKS AND DEFENCE STRATEGIES FOR IMPROVING ACCURACY IN MEDICAL IMAGE DIAGNOSIS

A **Report**
Submitted in Partial Fulfillment of the
Requirements for the Degree of
Bachelor of Technology
by

M Ajay Kumar Naidu (B210742EC), M Veera Abhi Nanda (B210708EC)
N Vijay Praneeth (B210658EC), Pranav Kandukuru (B210612EC)
Liyana N K (B210761EC)

under the supervision of

Dr. Ameer P M, NIT Calicut

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
NATIONAL INSTITUTE OF TECHNOLOGY CALICUT
NIT CAMPUS P.O. - 673601, KOZHIKODE, KERALA, INDIA

September 2024

Abstract

Deep learning has become indispensable in medical image analysis by significantly enhancing diagnostic accuracy and efficiency. Leveraging advanced neural networks, deep learning algorithms can process and analyse complex medical images—such as MRI, CT scans, and X-rays—with unparalleled precision. Although it has brought unprecedented advancements, it also heightened prediction unreliability, particularly from **adversarial attacks**. These attacks are designed to attack natural image classification models, which inevitably corrupt the semantics of pixels by applying spatial perturbations, which poses serious threat to the integrity of AI-driven diagnostic systems, potentially leading to erroneous clinical decisions.

In this project, we aim to explore and develop a deep-learning-based **defence mechanism** capable of identifying and mitigating these novel frequency-based adversarial attacks. Our initial approach involves training a **ResNet50** , **ResNet18** models using fully functional framework known as **PyTorch** on two public datasets - **Kvasir** and **Chest X-ray** datasets. We plan to implement these frequency-targeted adversarial attacks on the medical images from both datasets, followed by applying our idealogy behind the defence strategy, which is to remove perturbations in the higher frequency components. We will refine our defense mechanism by comparing original and defended image accuracy, aiming to enhance robustness against attacks and contribute to secure, accurate diagnostic systems for better healthcare.

Keywords :- Adversarial Attacks, Deep Learning, ResNet50, ResNet18, PyTorch

Contents

Chapter 1

Introduction

Deep learning has emerged as a transformative technology in the field of medical image analysis, offering unprecedented improvements in diagnostic accuracy and efficiency. By leveraging advanced neural networks, deep learning models can process and interpret complex medical images—such as MRI, CT scans, and X-rays—with remarkable precision. These capabilities have revolutionized the diagnostic process, enabling faster and more reliable analyses that ultimately improve patient outcomes and facilitate personalized treatment strategies. However, alongside these advancements, there are growing concerns about the vulnerability of deep learning models to adversarial attacks, particularly in the sensitive domain of medical imaging. These attacks can lead to significant prediction unreliability, potentially resulting in erroneous clinical decisions and compromising the integrity of AI-driven diagnostic systems.

1.1 Motivation

The increasing reliance on deep learning in medical diagnostics necessitates a robust understanding of the potential risks associated with adversarial attacks. Traditional adversarial attacks, designed for natural image classification models, pose a significant threat to medical image analysis by corrupting pixel semantics through spatial perturbations. This not only undermines the accuracy of diagnostic systems but also raises serious ethical and safety concerns. Given the critical role of AI in healthcare, there is an urgent need to develop effective defense mechanisms that can mitigate

these risks and ensure the reliability of diagnostic tools. Our motivation stems from the need to safeguard the integrity of AI-driven diagnostic systems, thereby enhancing the quality and safety of patient care.

1.2 Problem definition

The primary problem addressed by this project is the development of a robust and reliable defense mechanism that can protect deep learning models used in medical diagnostics from frequency-based adversarial attacks. These attacks exploit the model's sensitivity to high-frequency perturbations, potentially leading to dangerous misdiagnoses if not adequately defended against. The challenge lies in effectively identifying and mitigating these adversarial perturbations while maintaining the integrity of the original medical images, ensuring that the diagnostic accuracy of the deep learning models is preserved. This project seeks to bridge this gap by implementing a frequency-based defense strategy capable of safeguarding AI-driven diagnostic systems, thereby enhancing their security and reliability in clinical settings.

1.3 Objectives

The primary objective of this study is to explore and develop a deep-learning-based defense mechanism capable of identifying and mitigating frequency-based adversarial attacks in medical image analysis. Specifically, we aim to: Implement frequency-constrained adversarial attacks on medical images from the Kvasir and Chest X-ray datasets. Develop a defense strategy that targets higher frequency perturbations while preserving the lower frequency components of the images. Evaluate the effectiveness of the defense mechanism by comparing the prediction accuracy of original and defended images. Fine-tune the defense mechanism to enhance robustness against various types of adversarial attacks. Contribute to the development of more secure and accurate AI-driven diagnostic systems, supporting the delivery of high-quality healthcare.

1.4 Organization of the report

- **Chapter 1: Introduction** – Overview of the significance of deep learning in medical image analysis, challenges posed by adversarial attacks, and motivation behind this study.
- **Chapter 2: Literature Review** – Discussion of existing research on deep learning in medical imaging, adversarial attacks, and current defense mechanisms.
- **Chapter 3: Methodology** – Details on implementing frequency-constrained adversarial attacks, developing the defense mechanism, datasets used, and model architecture.
- **Chapter 4: Results** – Analysis of model performance against adversarial attacks, effectiveness of RPCA defense, and impact on classification accuracy.
- **Chapter 5: Work Plan** – Overview of ongoing tasks (RPCA refinement, performance evaluation) and future directions, including advanced defense strategies.
- **Chapter 6: Conclusion** – Summary of findings, implications for robust AI diagnostics, and potential for advancing defense strategies.

Chapter 2

Background

2.1 Deep Learning in Medical Image Analysis

Deep learning has revolutionized medical image analysis by providing automated, accurate, and efficient diagnostic capabilities. Advanced neural networks, such as Convolutional Neural Networks (CNNs) [1] and Deep Neural Networks (DNNs) [2], have demonstrated exceptional performance in tasks like image classification, segmentation, and detection across various medical imaging modalities including MRI, CT scans, and X-rays. These models can extract complex features from high-dimensional data, enabling precise identification and characterization of pathological conditions [3].

2.2 Vulnerabilities of Deep Learning Models to Adversarial Attacks

Despite their impressive capabilities, deep learning models are susceptible to adversarial attacks—intentional perturbations to input data that deceive models into making incorrect predictions. In medical contexts, such attacks can have severe consequences, potentially leading to misdiagnoses and inappropriate treatments. [4]

2.2.1 Traditional Adversarial Attacks

Traditional adversarial attacks on natural images often involve adding carefully crafted noise to input images, which can significantly alter model predictions while remaining imperceptible to human observers. Common techniques include:

- **Fast Gradient Sign Method (FGSM):** – Introduced by Goodfellow et al., FGSM generates adversarial examples by computing the gradient of the loss concerning the input image and adding a small perturbation in the direction that maximizes the loss [5].
- **Projected Gradient Descent (PGD):** – An iterative refinement of FGSM that applies multiple small perturbations, projecting the result back onto a feasible set after each step to maintain imperceptibility [5].
- **Carlini & Wagner (C&W) Attack:** – A powerful optimization-based attack that minimizes the perturbation added while ensuring misclassification [6].

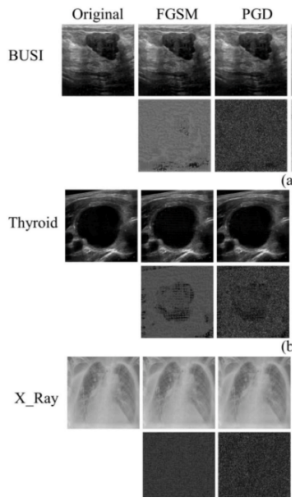

Fig. 2.1 The adversarial examples and perturbations generated by different attack methods on 2D medical image datasets with different modalities against ResNet18 model [4].

However, when these traditional attack methods are applied to medical images, they often introduce perturbations that, while subtle, can disrupt critical diagnostic features and compromise clinical decisions.

2.3 Frequency-Based Adversarial Attacks in Medical Imaging

To address the limitations of traditional attacks, recent research has focused on exploiting the frequency domain properties of images to create more effective and stealthy adversarial examples.

2.3.1 Frequency Constraint-Based Adversarial Attack

Chen et al. (2023) proposed a novel Frequency Constraint-Based Adversarial Attack tailored for medical image classification tasks. This method strategically injects perturbations into the high-frequency components of images while preserving the low-frequency content, which is typically associated with the primary structural and semantic information crucial for diagnosis. Key Features of the Approach [6]:

- **High-Frequency Perturbations:** – By constraining perturbations to high-frequency domains, the attack maintains visual similarity, making alterations imperceptible to human observers and less detectable by standard defense mechanisms.
- **Feature Space Attack:** – The method operates in the feature representation space rather than directly altering output logits, enhancing the transferability of adversarial examples across different models and datasets.
- **Applicability Across Modalities:** – The attack has been tested on various medical imaging modalities and dimensionalities, including 2D chest X-rays, breast and thyroid ultrasound images, and 3D CT scans, demonstrating its versatility.

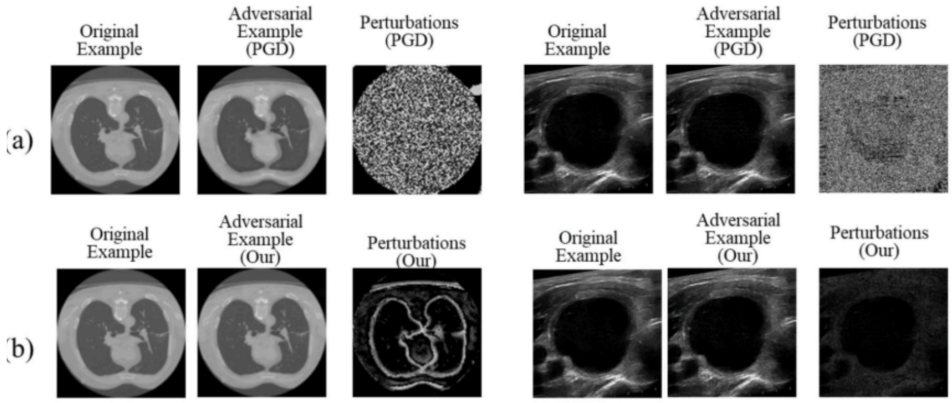

Fig. 2.2 Comparison of the original and adversarial examples generated by different attack methods. (a) Adversarial examples and perturbations generated by PGD method for CT and ultrasound medical images; (b) Adversarial examples and perturbations generated by our proposed method for CT and ultrasound medical images. For the visualization, we regularize the perturbations by taking its absolute value and multiplying it by 20 [6].

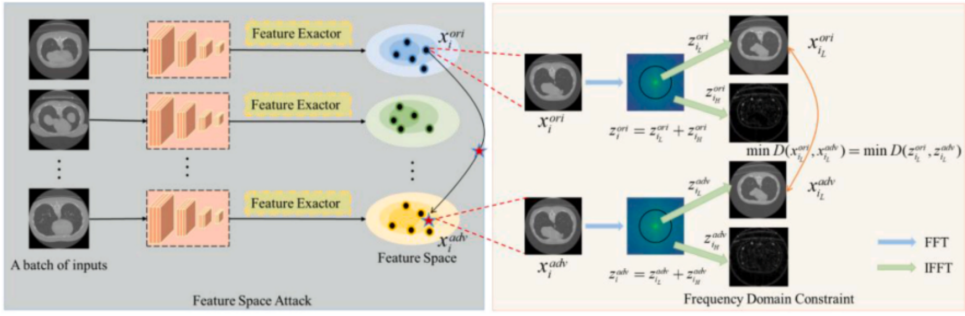

Fig. 2.3 The overview framework of the frequency constraint-based adversarial attack method. Left: Feature Space Attack; Right: frequency Domain Constraint. D() is the distance between original examples and corresponding adversarial examples in low-frequency components [6] .

Experimental Validation:

The proposed method was evaluated on four public medical image datasets:

- **Chest X-Ray Dataset:** – Demonstrated high attack success rates with minimal perceptual distortion, effectively misleading classification models while preserving image quality.
- **Breast Ultrasound Dataset (BUSI):** – Achieved superior performance over traditional attacks, maintaining the diagnostic integrity of images.
- **Thyroid Ultrasound Dataset:** – Showed robustness in generating effective adversarial examples across different tissue textures and structures.
- **3D CT Scan Dataset (Mosmed-1110):** – Extended applicability to volumetric data, successfully crafting adversarial examples in a high-dimensional space.

Performance Metrics:

- **Attack Success Rate (ASR):** – The method consistently achieved high ASR across all datasets, indicating its effectiveness in deceiving various deep learning models.
- **Frechet Inception Distance (FID):** – Low FID scores were observed, reflecting high similarity between original and adversarial images and confirming the imperceptibility of perturbations.
- **Low-Frequency Component Distortion (LF):** – Minimal LF values indicated that essential structural information was preserved, ensuring that adversarial examples remained realistic and clinically plausible.

Advantages over Traditional Methods:

- **Enhanced Imperceptibility:** By targeting high-frequency components, the attack avoids noticeable distortions, reducing the likelihood of detection by visual inspection or simple filtering techniques.

- **Improved Transferability:** Operating in the feature space allows adversarial examples to generalize across different models and datasets, posing a broader threat to various medical imaging systems.
- **Applicability to Diverse Modalities:** The method's success across multiple imaging modalities underscores its versatility and potential impact on a wide range of medical applications.

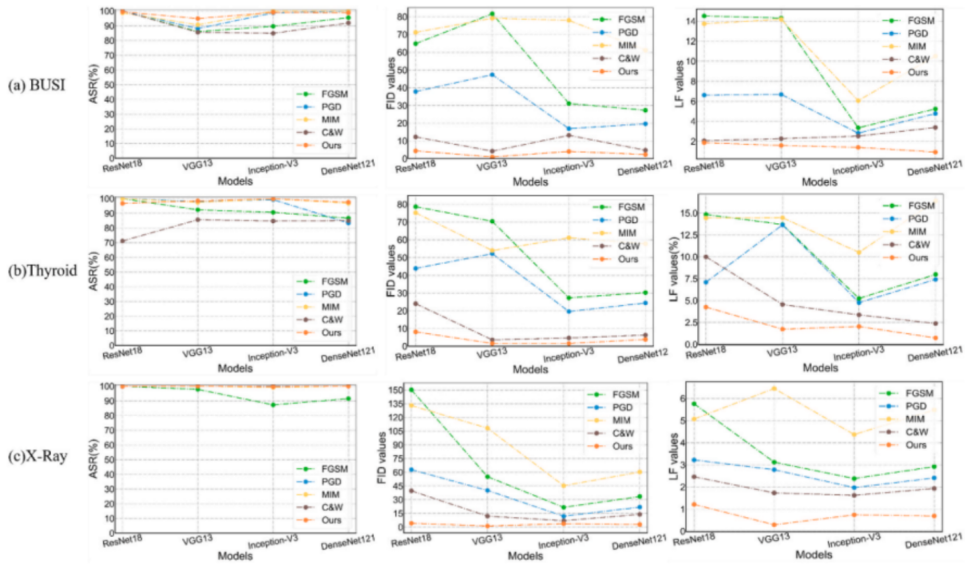

Fig. 2.4 The attack success rate, FID and LF values for different2D datasets against different models: (a) BUSI dataset, (b) Thyroid dataset and (c) X-Ray dataset. The left column presents the ASR, the middle column represents the FID value, the right column presents the LF value [6] .

2.4 Defense Mechanisms Against Frequency-Based Adversarial Attacks

The emergence of sophisticated frequency-based attacks necessitates the development of robust defense strategies tailored to counteract these specific threats.

2.4.1 Existing Defense Approaches

- **Adversarial Training:** Incorporating adversarial examples into the training process to enhance model robustness. However, this approach can be computationally intensive and may not generalize well to unforeseen attack strategies.
- **Input Transformation:** Applying preprocessing steps such as smoothing or denoising filters to mitigate perturbations. While effective against some attacks, these methods may degrade image quality and diagnostic information.
- **Frequency Domain Analysis:** Implementing defenses that specifically target high-frequency anomalies by analyzing the spectral properties of input images.

2.4.2 Proposed Defense Strategies

Building upon the understanding of frequency-based attacks, defense mechanisms can be developed that:

- **Selective Frequency Filtering:** Designing adaptive filters that target and suppress suspicious high-frequency patterns without compromising essential image details.
- **Frequency-Aware Model Architectures:** Incorporating modules within neural networks that explicitly analyze and validate frequency components, enhancing detection and resilience against adversarial perturbations.
- **Hybrid Approaches:** Combining multiple defense strategies, such as integrating frequency domain analysis with adversarial training, to provide layered and robust protection.

Challenges and Future Directions

- **Balancing Robustness and Accuracy:** Ensuring that defensive measures do not adversely affect the model's performance on clean, unperturbed data.
- **Real-Time Detection:** Developing efficient algorithms capable of identifying and mitigating adversarial attacks in real-time clinical settings.

- **Generalization Across Attacks:** Creating defenses that are effective against a wide spectrum of attack methods, including those yet to be developed.

Chapter 3

Methodology

3.1 Data Collection

The methodology of our study begins with the collection of relevant data from public datasets, namely the Kvasir and Chest X-ray datasets. These datasets were chosen due to their widespread use in medical imaging and the variety of medical conditions they encompass, providing a comprehensive basis for training and evaluating deep learning models.

Kvasir Dataset: This dataset contains images of the gastrointestinal tract, collected using endoscopic techniques. The dataset includes annotated images of several gastrointestinal diseases, making it ideal for training models in the context of gastrointestinal diagnostics.

Chest X-ray Dataset: This dataset includes thousands of chest X-ray images, with annotations for conditions like pneumonia, tuberculosis, and other lung diseases. The diversity and quantity of images make it suitable for studying the application of deep learning in thoracic disease detection.

3.2 Data Preprocessing

Data preprocessing is a critical step to ensure the quality and relevance of the input data, particularly when working with medical images.

3.2.1 Image Normalization and Augmentation

To enhance the performance of the deep learning models, images from both datasets are normalized to a common scale [7]. Image augmentation techniques such as rotation, flipping, and zooming are applied to increase the dataset's variability, which helps in improving the robustness of the model against real-world scenarios.

3.3 Model Implementation

The core of our methodology involves implementing and training deep learning models capable of both performing medical image classification and defending against adversarial attacks.

3.3.1 ResNet50 and ResNet18 Architecture

We utilized the ResNet50 and ResNet18 architecture, a deep convolutional neural network known for its ability to handle vanishing gradients through residual learning. The model is implemented using the PyTorch framework, and it is fine-tuned on the preprocessed Kvasir (here ResNet50 is used because the datasets has rgb images), Chest X-ray and BUSI datasets (here ResNet18 is used because it has gray images).

3.3.2 Adversarial Attack Implementation

The frequency-constrained adversarial attacks are implemented on the trained ResNet50 model [8]. These attacks focus on modifying the high-frequency components of the images, while preserving the low-frequency content, which typically carries the essential diagnostic information. The attacks are evaluated based on their ability to deceive the model without introducing perceptible changes to the images.

3.4 Frequency Based Adversial Attack

3.4.1 Feature Space Attack

In this section, the paper describes an approach to adversarial attacks targeting the feature space of images [9]. Instead of directly modifying pixel values to mislead

the classifier, this method manipulates the image's representation in feature space to increase similarity with a target class or decrease similarity with its original class.

Here's a breakdown of the approach, focusing on the key elements and equations involved:

1. **Feature Mapping Function** $M(\cdot)$:

   - A mapping function M is used to transform the input images into a feature space. This transformation captures the underlying features of each image, which enables the attack to target deeper semantic similarities rather than just pixel-level details.

2. **Optimization Objective**:

   - Given a batch of N samples $X = \{x_1, x_2, \ldots, x_N\}$, the goal is to find adversarial images x_i^{adv} that minimize the similarity to the original feature representation while maximizing dissimilarity to other samples in the batch.
   - This can be formalized as an optimization problem:

$$x_i^{\text{adv}} = \arg\min_{x_i'} \left\{ \max\left[0, h_{i,i} - \min\left(h_{i,j} \mid j \neq i\right)\right] \right\}$$

     Here:

     - x_i' is initialized as the original image x_i.
     - $h_{i,i} = \text{sim}(M(x_i'), M(x_i))$ denotes the similarity between the adversarial image and its original version.
     - $h_{i,j} = \text{sim}(M(x_i'), M(x_j))$ represents the similarity between the adversarial image and other images in the batch.

3. **Similarity Measure Using Cosine Similarity**:

   - The similarity $\text{sim}(M(x_i'), M(x_j))$ is computed using cosine similarity, which is suitable for comparing visual and semantic features between images. The cosine similarity equation is given by:

$$\text{sim}(M(x_i'), M(x_j)) = \frac{M(x_i')^T M(x_j)}{\|M(x_i')\|_2 \cdot \|M(x_j)\|_2}$$

This measure helps identify how closely the features of the adversarial image align with those of the original or other images.

4. **Targeted Attack Scenario**:

   - For a targeted attack, where the adversarial image is intended to resemble a specific target category t, the optimization objective modifies as follows:

$$x_i^{\text{adv}} = \arg\min_{x_i'} \{\max[0, h_{i,i} - h_{i,t}]\}$$

   where $h_{i,t}$ denotes the similarity between the adversarial image and a target image feature in the batch.

5. **Non-Targeted Attack Strategy**:

   - In a non-targeted attack, the goal is to reduce the similarity between the adversarial and original images. The adversarial image is optimized to be less like its original representation and more similar to dissimilar examples in the batch, pushing it into a different feature subspace.

This approach enables adversarial images to blend effectively in feature space, potentially misleading the classifier by moving the representation toward the target or away from the original classification [10].

3.4.2 Frequency Domain Constraint

In Section 3.2, "Frequency Domain Constraint," the paper discusses an approach to restrict adversarial perturbations to high-frequency components in the frequency domain. The aim is to make adversarial modifications less noticeable to human perception while still affecting the model's predictions. Here's a step-by-step breakdown:

1. **Fourier Transform to Frequency Domain**:

- The adversarial attack begins by transforming the image into the frequency domain using the Fourier Transform [11]. This transformation allows the separation of low-frequency (smooth and broad details) and high-frequency (edges and fine textures) components.
- For an image $f(x,y)$, its Fourier Transform $F(u,v)$ is calculated as:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \left(\cos\left(2\pi \frac{ux}{M} + \frac{vy}{N}\right) - i \sin\left(2\pi \frac{ux}{M} + \frac{vy}{N}\right) \right)$$

  where:
  - $F(u,v)$ represents the frequency spectrum of the image.
  - M and N are the image dimensions.
  - u and v represent frequency indices.
  - The result $F(u,v)$ is complex, capturing both amplitude and phase of the frequencies.

2. **Separating Low and High-Frequency Components**:
   - After computing the Fourier Transform, the spectrum $F(u,v)$ is divided into low and high-frequency components. This separation is achieved by setting a radius threshold r:
     - Points within a distance r from the center of the frequency spectrum are considered low-frequency components.
     - Points beyond r are considered high-frequency components.
   - The decomposition functions are defined as:

$$z_L(i,j) = \begin{cases} z(i,j), & \text{if } d((i,j),(c_i,c_j)) \leq r \\ 0, & \text{otherwise} \end{cases}$$

$$z_H(i,j) = \begin{cases} 0, & \text{if } d((i,j),(c_i,c_j)) \leq r \\ z(i,j), & \text{otherwise} \end{cases}$$

   where $d((i,j),(c_i,c_j))$ is the Euclidean distance from the center.

3. **Low-Frequency Distance Constraint**:

   - To maintain similarity between the adversarial and original images in the low-frequency spectrum, the distance between their low-frequency components is minimized. The distance D_L is computed as:

$$D_L(z_L^{\text{ori}}, z_L^{\text{adv}}) = d((a^{\text{ori}}, b^{\text{ori}}), (a^{\text{adv}}, b^{\text{adv}}))$$

     where $d(\cdot,\cdot)$ is a distance metric, like the Euclidean distance, between complex values of the low-frequency components.

4. **Final Objective Function with Frequency Domain Constraint**:

   - The frequency domain constraint is integrated into the overall objective function to guide the optimization of adversarial examples. The final objective is:

$$x_i^{\text{adv}} = \arg\min_{x_i'} \left\{ \max\left[0, \alpha_i h_{i,i} - \beta_i \min\left(h_{i,j} \mid j \neq i\right)\right] + \lambda D_L \right\}$$

     where:

     – α_i and β_i are weights to balance the similarity scores.
     – λ is a hyperparameter controlling the importance of the low-frequency distance constraint D_L.

   - This objective ensures that adversarial perturbations are concentrated in the high-frequency regions while minimizing changes to the low-frequency components, enhancing the imperceptibility of the adversarial image.

The frequency domain constraint technique allows the adversarial perturbations

3.4.3 Explanation of the Algorithm

Based on the explanations of the **Feature Space Attack** (Section 3.4.1) and **Frequency Domain Constraint** (Section 3.4.2), here's how the algorithm works.

Algorithm 1 Adversarial attack by using the proposed method

Input: The map function M(·)(feature exactor); a batch of original images $\{x_i^{ori}\}_{i=1}^{N}$; the number of iterations K.

Output: A batch of adversarial images $\{x_i^{adv}\}_{i=1}^{N}$

Initialize: $\{x_i^{adv}\}_{i=1}^{N} \neg \{x_i^{ori}\}_{i=1}^{N}$

for $i = 1$ to N:

  $r_i \neg \text{arctanh}(2x_i^{adv} - 1)$

  for $k = 1$ to K:

    $h_{i,i} \neg$ Calcuate $sim(M(r_i'), M(r_i))$ from i to N

    $h_{i,j} \neg$ Calcuate $sim(M(r_i'), M(r_j))(j \neq i)$

    $z_L^{ori}, z_L^{adv} \neg$ Use Fourier transform get low frequency from x_i^{ori}, x_i^{adv}

    $D_L \neg d(z_L^{ori}, z_L^{adv})$

    $r_i \neg$ Optimize variable r_i by $\arg\min_{x_i} \{\max[0, \alpha_i h_{i,j} - \beta_i \min(h_{i,j} \mid j \neq i)]\} + \lambda D_L$

    $x_i^{adv} \neg \frac{1}{2}\tanh(r_i)$

  end for

end for

return $\{x_i^{adv}\}_{i=1}^{N}$

Fig. 3.1 Algorithm for generating adversarial examples using feature space attack and frequency domain constraint

3.4.4 Overview of the Algorithm

This algorithm generates adversarial examples by combining a **feature space attack** (which alters the image representation in the feature space to mislead the classifier) with a **frequency domain constraint** (which limits perturbations to high-frequency regions, ensuring they're less visible to the human eye).

Steps in the Algorithm

1. **Input and Initialization**:
   - The inputs are:
     - $M(\cdot)$: A feature extraction function (often a deep learning model that maps images to a feature space).
     - $\{x_i^{\text{ori}}\}_{i=1}^{N}$: A batch of original images.

  - K: Number of iterations for the attack.
- The algorithm initializes each adversarial example x_i^{adv} as a copy of its original counterpart x_i^{ori}.

2. **Outer Loop Over Each Image in the Batch**:
   - For each image x_i^{adv} in the batch, the algorithm proceeds to generate an adversarial version by performing iterative updates.
3. **Initialization for Each Image r_i**:
   - The variable r_i is initialized using the `arctanh` transformation applied to x_i^{adv}. This helps to keep the adversarial modifications within a bounded range, ensuring that changes to pixel values remain within allowable limits.
4. **Inner Loop for Iterative Optimization (Up to K Iterations)**:
   - Each iteration aims to make x_i^{adv} closer in feature space to a target (or less similar to its original class) while enforcing the frequency constraint.
   - **Step 4.1: Similarity Calculation**:
     - For each other image x_j in the batch, the algorithm calculates the similarity $h_{i,j} = \text{sim}(M(r_i), M(x_j))$.
     - This similarity, based on cosine distance, helps measure how close x_i^{adv} is to other classes in the feature space.
   - **Step 4.2: Fourier Transform and Low-Frequency Extraction**:
     - The algorithm computes the Fourier transform for both x_i^{ori} and x_i^{adv}, obtaining their respective frequency spectra.
     - Low-frequency components (representing broad, structural details) are separated from high-frequency components (representing textures and edges). This is done using a threshold radius r around the center of the Fourier spectrum.
   - **Step 4.3: Low-Frequency Distance Calculation D_L**:

  - A distance D_L is computed between the low-frequency components of x_i^{ori} and x_i^{adv}. This measures the difference in essential structural information between the original and adversarial images, ensuring that major structures are preserved.
- **Step 4.4: Optimization Update for r_i**:
  - The algorithm updates r_i by minimizing the following combined objective:
    * **Maximizing dissimilarity** to the original class: Reduces $h_{i,i}$ (similarity with its original feature) and increases $h_{i,j}$ (similarity with other images).
    * **Frequency domain constraint**: Adds λD_L, ensuring that the low-frequency component remains close to the original.
  - This update encourages the adversarial example x_i^{adv} to have minimal detectable changes while effectively attacking the feature space.

5. **Update the Adversarial Image x_i^{adv}**:
   - After K iterations, r_i is transformed back, and x_i^{adv} is updated. The result is an adversarial image that is close to the original in low frequencies (imperceptible changes) but differs enough in feature space to mislead the classifier.

6. **Output**:
   - The algorithm returns a batch of adversarial images $\{x_i^{\text{adv}}\}_{i=1}^{N}$, each optimized to attack the classifier while appearing visually similar to the original.

3.4.5 Summary

The algorithm effectively combines:

- A **feature space attack**, making the adversarial image less similar to its original class in feature space.

- A **frequency domain constraint**, keeping modifications in high-frequency regions, which are less perceptible to human vision.

This two-pronged approach ensures the adversarial examples are both effective in misleading the classifier and stealthy in terms of visual appearance.

3.5 Defense Mechanism Development

The defense mechanism we are developing in this project leverages Robust Principal Component Analysis (RPCA), specifically targeting the high-frequency components of medical images to counteract adversarial perturbations.

3.5.1 RPCA-Based High-Frequency Component Filtering

This strategy involves decomposing the image into its singular value decomposition (SVD) matrix using RPCA. By analyzing the singular values, the method identifies and isolates high-frequency components that are likely to contain adversarial perturbations. The defense mechanism then iteratively refines these components by introducing Gaussian noise, which helps to neutralize the perturbations without significantly altering the essential image content. This iterative process continues until the adversarial effects are sufficiently mitigated, ensuring that the image remains diagnostically accurate while being robust against attacks.

3.5.2 Working of the mechanism

1. **Adding Gaussian Noise (`add_gaussian_noise`)**:
    - This function applies Gaussian noise to an image, which can help create variations in the input data that may weaken certain adversarial perturbations, making the attack less effective.
    - `stddev` controls the amount of noise. Gaussian noise is added independently to each color channel.
2. **Frobenius Norm Calculation (`frobeniusNorm`)**:

- Calculates the Frobenius norm of a matrix, which is a measure of the magnitude of elements in the matrix. This is used to quantify convergence by comparing the differences between the original and decomposed matrices.

3. **Convergence Check (`converged`)**:

   - Computes the relative error between the original matrix H and the sum of the decomposed components J, S, and W.
   - The function stops iterating once this error is below a small threshold (e.g., $1e-6$), indicating that the RPCA algorithm has reached a stable state.

4. **Soft Thresholding (`shrink`)**:

   - Implements soft thresholding, which is applied element-wise to the matrix. Elements are shrunk towards zero by a threshold τ. This is often used to isolate small noise components while preserving the main structure in a matrix.

5. **SVD-Based Soft Thresholding (`svd_shrink`)**:

   - This function uses Singular Value Decomposition (SVD) to separate the low-rank component by applying soft thresholding to the singular values of the matrix X. The SVD decomposition is helpful for distinguishing between important structural information and noise.
   - U, s, and V are derived from SVD, where s (singular values) are adjusted by the shrink function to emphasize low-rank features and suppress noise.

6. **RPCA Decomposition (`RPCA_N`)**:

   - This function decomposes the input matrix H (which could represent an image or a feature matrix) into three components:
     - J: the low-rank component representing the primary structure or background of the image.

  - S: the sparse component capturing small, high-frequency details and edges, including potential adversarial perturbations.
  - W: the noise component, which isolates high-frequency information.
- The decomposition is performed iteratively, with the parameters λ, β, μ, max_mu, and ρ controlling the balance between convergence speed and separation quality.
- The Lagrange multipliers G_1 and G_2 are updated in each iteration to minimize the difference between the decomposed components and the original matrix H.

3.5.3 RPCA as a Defense Mechanism

RPCA effectively decomposes an image into structured (low-rank) and unstructured (sparse and noise) components. Here's how it works as a defense mechanism against frequency-based adversarial attacks:

1. **Separation of High-Frequency Perturbations**:
   - Frequency-based adversarial attacks tend to embed perturbations in high-frequency regions (e.g., edges or textures) to make them less perceptible. In RPCA, the sparse component S and noise component W capture these high-frequency elements, effectively isolating potential adversarial perturbations from the primary image structure in J.
2. **Preservation of the Low-Frequency Component**:
   - The low-rank component J represents the main structure of the image, capturing low-frequency information and discarding high-frequency noise. By isolating J, the model can focus on the primary content of the image, reducing the influence of high-frequency adversarial modifications that might be present in S or W.
3. **Filtering Out Adversarial Noise**:
   - Once RPCA decomposition is complete, the reconstructed image can be formed by retaining only the low-rank component J. This effectively

removes sparse and noise elements (potentially containing adversarial perturbations) from the image, thus weakening the attack. The sparse and noise components S and W may contain high-frequency adversarial patterns that contribute minimally to the main structure.

4. **Defensive Reconstruction**:

   - By using only the J component for classification or further processing, the system can achieve higher robustness against frequency-based attacks. This reconstruction filters out high-frequency adversarial noise, allowing the model to operate on a "cleaner" version of the image.

3.5.4 Model Evaluation and Fine-Tuning

The effectiveness of the RPCA-based defense mechanism is evaluated by comparing the classification accuracy of both the original and defended images. Fine-tuning of the defense strategy is performed to optimize its performance across various types of adversarial attacks, ensuring minimal impact on the diagnostic quality of the images.

Chapter 4

Results

4.1 Model Implementation Results

The Kvasir dataset images are trained with ResNet50 architecture with an accuracy of 93.8% and the chest X-ray, BUSI datasets are trained with ResNet18 architecture with an accuracy of 94.36% and 92.18% respectively.

4.1.1 Traditional Adversarial Attack Implementation Result

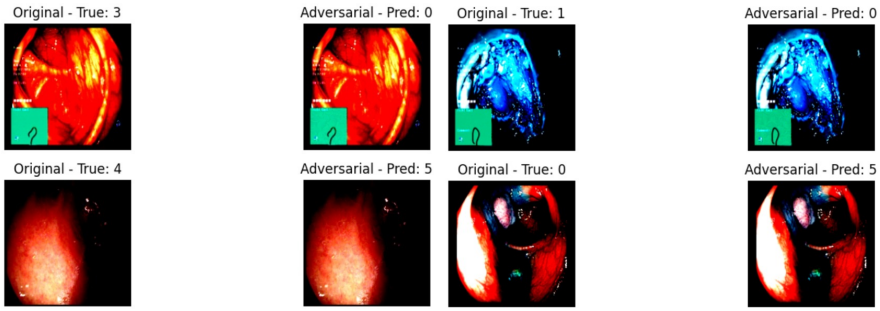

Fig. 4.1 Original images with predicted labels and corresponding Attacked images with prediction by the model

```
FGSM Attack:
Success Rate: 89.38%
Image 0: Original: 3, Adversarial: 5
Image 1: Original: 4, Adversarial: 5
Image 2: Original: 1, Adversarial: 1
Image 3: Original: 0, Adversarial: 0
Image 4: Original: 2, Adversarial: 5
.
.
.
Image 635: Original: 1, Adversarial: 0
Image 636: Original: 3, Adversarial: 0
Image 637: Original: 7, Adversarial: 0
Image 638: Original: 5, Adversarial: 2
Image 639: Original: 1, Adversarial: 5
```

```
PGD Attack:
Success Rate: 99.84%
Image 0: Original: 3, Adversarial: 0
Image 1: Original: 4, Adversarial: 5
Image 2: Original: 1, Adversarial: 0
Image 3: Original: 0, Adversarial: 5
Image 4: Original: 2, Adversarial: 1
.
.
.
Image 635: Original: 1, Adversarial: 0
Image 636: Original: 3, Adversarial: 0
Image 637: Original: 7, Adversarial: 0
Image 638: Original: 5, Adversarial: 2
Image 639: Original: 1, Adversarial: 5
```

```
BIM Attack:
Success Rate: 98.12%
Image 0: Original: 3, Adversarial: 0
Image 1: Original: 4, Adversarial: 5
Image 2: Original: 1, Adversarial: 0
Image 3: Original: 0, Adversarial: 1
Image 4: Original: 2, Adversarial: 5
.
.
.
Image 635: Original: 1, Adversarial: 0
Image 636: Original: 3, Adversarial: 0
Image 637: Original: 7, Adversarial: 6
Image 638: Original: 5, Adversarial: 2
Image 639: Original: 1, Adversarial: 0
```

```
DeepFool Attack:
Success Rate: 75.94%
Image 0: Original: 3, Adversarial: 0
Image 1: Original: 4, Adversarial: 5
Image 2: Original: 1, Adversarial: 1
Image 3: Original: 0, Adversarial: 0
Image 4: Original: 2, Adversarial: 2
.
.
.
Image 635: Original: 1, Adversarial: 0
Image 636: Original: 3, Adversarial: 0
Image 637: Original: 7, Adversarial: 0
Image 638: Original: 5, Adversarial: 2
Image 639: Original: 1, Adversarial: 1
```

```
C&W Attack:
Success Rate: 99.53%
Image 0: Original: 3, Adversarial: 0
Image 1: Original: 4, Adversarial: 5
Image 2: Original: 1, Adversarial: 0
Image 3: Original: 0, Adversarial: 1
Image 4: Original: 2, Adversarial: 1
.
.
.
Image 635: Original: 1, Adversarial: 5
Image 636: Original: 3, Adversarial: 0
Image 637: Original: 7, Adversarial: 0
Image 638: Original: 5, Adversarial: 2
Image 639: Original: 1, Adversarial: 5
```

Fig. 4.2 (a) FGSM Attack (b) PGD Attack (c) BIM Attack (d) Deepfool Attack (e) C&W Attack

4.1.2 Results after Using the Defense Mechanism on Attacked Images using RPCA

```
Original Labels | Noisy Predictions | RPCA Predictions
Image 1: Original: 3 | Noisy: 4 | RPCA: 3
Image 2: Original: 2 | Noisy: 5 | RPCA: 2
Image 3: Original: 2 | Noisy: 5 | RPCA: 2
Image 4: Original: 4 | Noisy: 5 | RPCA: 4
Image 5: Original: 6 | Noisy: 1 | RPCA: 0
Image 6: Original: 7 | Noisy: 5 | RPCA: 7
Image 7: Original: 7 | Noisy: 5 | RPCA: 7
Image 8: Original: 5 | Noisy: 5 | RPCA: 5
Image 9: Original: 6 | Noisy: 5 | RPCA: 6
Image 10: Original: 4 | Noisy: 5 | RPCA: 4
Image 11: Original: 4 | Noisy: 5 | RPCA: 4
Image 12: Original: 6 | Noisy: 5 | RPCA: 6
Image 13: Original: 6 | Noisy: 5 | RPCA: 6
Image 14: Original: 1 | Noisy: 1 | RPCA: 1
Image 15: Original: 1 | Noisy: 1 | RPCA: 1
Image 16: Original: 4 | Noisy: 5 | RPCA: 4
Image 17: Original: 7 | Noisy: 5 | RPCA: 7
Image 18: Original: 6 | Noisy: 5 | RPCA: 6
Image 19: Original: 4 | Noisy: 4 | RPCA: 4
Image 20: Original: 0 | Noisy: 5 | RPCA: 0
Image 21: Original: 0 | Noisy: 5 | RPCA: 0
Image 22: Original: 0 | Noisy: 0 | RPCA: 0
Image 23: Original: 1 | Noisy: 5 | RPCA: 1
Image 24: Original: 7 | Noisy: 5 | RPCA: 7
Image 25: Original: 0 | Noisy: 5 | RPCA: 1
Image 26: Original: 4 | Noisy: 5 | RPCA: 4
Image 27: Original: 2 | Noisy: 5 | RPCA: 2
Image 28: Original: 2 | Noisy: 4 | RPCA: 2
Image 29: Original: 5 | Noisy: 5 | RPCA: 5
Image 30: Original: 4 | Noisy: 4 | RPCA: 4
Image 31: Original: 5 | Noisy: 5 | RPCA: 5
Image 32: Original: 1 | Noisy: 5 | RPCA: 1
Image 33: Original: 7 | Noisy: 5 | RPCA: 7
Image 34: Original: 3 | Noisy: 5 | RPCA: 3
Image 35: Original: 0 | Noisy: 0 | RPCA: 0
```

(a) RPCA Results on original Images with original predictions with retention = 92.57%

```
Original Labels | Attacked Predictions | RPCA Predictions
Image 1: Original: 3 | Attacked: 0 | RPCA: 7
Image 2: Original: 2 | Attacked: 5 | RPCA: 2
Image 3: Original: 2 | Attacked: 5 | RPCA: 2
Image 4: Original: 4 | Attacked: 0 | RPCA: 0
Image 5: Original: 6 | Attacked: 0 | RPCA: 6
Image 6: Original: 7 | Attacked: 0 | RPCA: 6
Image 7: Original: 7 | Attacked: 0 | RPCA: 0
Image 8: Original: 5 | Attacked: 2 | RPCA: 2
Image 9: Original: 6 | Attacked: 5 | RPCA: 7
Image 10: Original: 4 | Attacked: 5 | RPCA: 7
Image 11: Original: 4 | Attacked: 5 | RPCA: 7
Image 12: Original: 6 | Attacked: 0 | RPCA: 6
Image 13: Original: 6 | Attacked: 0 | RPCA: 6
Image 14: Original: 1 | Attacked: 0 | RPCA: 0
Image 15: Original: 1 | Attacked: 5 | RPCA: 6
Image 16: Original: 4 | Attacked: 5 | RPCA: 4
Image 17: Original: 7 | Attacked: 5 | RPCA: 7
Image 18: Original: 6 | Attacked: 0 | RPCA: 6
Image 19: Original: 4 | Attacked: 5 | RPCA: 2
Image 20: Original: 0 | Attacked: 5 | RPCA: 0
Image 21: Original: 0 | Attacked: 1 | RPCA: 0
Image 22: Original: 0 | Attacked: 5 | RPCA: 0
Image 23: Original: 1 | Attacked: 5 | RPCA: 0
Image 24: Original: 7 | Attacked: 0 | RPCA: 7
Image 25: Original: 0 | Attacked: 1 | RPCA: 6
Image 26: Original: 4 | Attacked: 5 | RPCA: 0
Image 27: Original: 2 | Attacked: 5 | RPCA: 2
Image 28: Original: 2 | Attacked: 5 | RPCA: 2
Image 29: Original: 5 | Attacked: 2 | RPCA: 2
Image 30: Original: 4 | Attacked: 5 | RPCA: 2
Image 31: Original: 5 | Attacked: 2 | RPCA: 2
Image 32: Original: 1 | Attacked: 0 | RPCA: 1
Image 33: Original: 7 | Attacked: 0 | RPCA: 7
Image 34: Original: 3 | Attacked: 0 | RPCA: 7
Image 35: Original: 0 | Attacked: 1 | RPCA: 0
```

(b) RPCA results of images attacked with PGD

Fig. 4.3 RPCA results on original images and images attacked with PGD

```
Original Labels | Attacked Predictions | RPCA Predictions
Image 1: Original: 3 | Attacked: 5 | RPCA: 7
Image 2: Original: 2 | Attacked: 4 | RPCA: 2
Image 3: Original: 2 | Attacked: 5 | RPCA: 2
Image 4: Original: 4 | Attacked: 0 | RPCA: 0
Image 5: Original: 6 | Attacked: 0 | RPCA: 6
Image 6: Original: 7 | Attacked: 0 | RPCA: 6
Image 7: Original: 7 | Attacked: 0 | RPCA: 0
Image 8: Original: 5 | Attacked: 2 | RPCA: 2
Image 9: Original: 6 | Attacked: 5 | RPCA: 7
Image 10: Original: 4 | Attacked: 4 | RPCA: 7
Image 11: Original: 4 | Attacked: 5 | RPCA: 7
Image 12: Original: 6 | Attacked: 0 | RPCA: 6
Image 13: Original: 6 | Attacked: 0 | RPCA: 6
Image 14: Original: 1 | Attacked: 0 | RPCA: 0
Image 15: Original: 1 | Attacked: 0 | RPCA: 6
Image 16: Original: 4 | Attacked: 5 | RPCA: 4
Image 17: Original: 7 | Attacked: 5 | RPCA: 7
Image 18: Original: 6 | Attacked: 4 | RPCA: 6
Image 19: Original: 4 | Attacked: 5 | RPCA: 2
Image 20: Original: 0 | Attacked: 0 | RPCA: 0
Image 21: Original: 0 | Attacked: 1 | RPCA: 0
Image 22: Original: 0 | Attacked: 0 | RPCA: 0
Image 23: Original: 1 | Attacked: 1 | RPCA: 0
Image 24: Original: 7 | Attacked: 0 | RPCA: 7
Image 25: Original: 0 | Attacked: 1 | RPCA: 6
Image 26: Original: 4 | Attacked: 2 | RPCA: 0
Image 27: Original: 2 | Attacked: 5 | RPCA: 2
Image 28: Original: 2 | Attacked: 5 | RPCA: 2
Image 29: Original: 5 | Attacked: 2 | RPCA: 2
Image 30: Original: 4 | Attacked: 0 | RPCA: 2
Image 31: Original: 5 | Attacked: 2 | RPCA: 2
Image 32: Original: 1 | Attacked: 0 | RPCA: 1
Image 33: Original: 7 | Attacked: 0 | RPCA: 7
Image 34: Original: 3 | Attacked: 5 | RPCA: 7
Image 35: Original: 0 | Attacked: 0 | RPCA: 0
```

(a) RPCA results of images attacked with FGSM

```
Original Labels | Attacked Predictions | RPCA Predictions
Image 1: Original: 3 | Attacked: 6 | RPCA: 7
Image 2: Original: 2 | Attacked: 5 | RPCA: 2
Image 3: Original: 2 | Attacked: 5 | RPCA: 2
Image 4: Original: 4 | Attacked: 5 | RPCA: 0
Image 5: Original: 6 | Attacked: 0 | RPCA: 6
Image 6: Original: 7 | Attacked: 2 | RPCA: 6
Image 7: Original: 7 | Attacked: 0 | RPCA: 0
Image 8: Original: 5 | Attacked: 2 | RPCA: 2
Image 9: Original: 6 | Attacked: 7 | RPCA: 7
Image 10: Original: 4 | Attacked: 3 | RPCA: 7
Image 11: Original: 4 | Attacked: 5 | RPCA: 7
Image 12: Original: 6 | Attacked: 0 | RPCA: 6
Image 13: Original: 6 | Attacked: 2 | RPCA: 6
Image 14: Original: 1 | Attacked: 0 | RPCA: 0
Image 15: Original: 1 | Attacked: 1 | RPCA: 6
Image 16: Original: 4 | Attacked: 5 | RPCA: 4
Image 17: Original: 7 | Attacked: 6 | RPCA: 7
Image 18: Original: 6 | Attacked: 4 | RPCA: 6
Image 19: Original: 4 | Attacked: 5 | RPCA: 0
Image 20: Original: 0 | Attacked: 1 | RPCA: 0
Image 21: Original: 0 | Attacked: 1 | RPCA: 0
Image 22: Original: 0 | Attacked: 3 | RPCA: 0
Image 23: Original: 1 | Attacked: 0 | RPCA: 0
Image 24: Original: 7 | Attacked: 6 | RPCA: 7
Image 25: Original: 0 | Attacked: 1 | RPCA: 6
Image 26: Original: 4 | Attacked: 5 | RPCA: 0
Image 27: Original: 2 | Attacked: 5 | RPCA: 2
Image 28: Original: 2 | Attacked: 5 | RPCA: 2
Image 29: Original: 5 | Attacked: 2 | RPCA: 2
Image 30: Original: 4 | Attacked: 6 | RPCA: 2
Image 31: Original: 5 | Attacked: 2 | RPCA: 2
Image 32: Original: 1 | Attacked: 0 | RPCA: 1
Image 33: Original: 7 | Attacked: 0 | RPCA: 7
Image 34: Original: 3 | Attacked: 6 | RPCA: 7
Image 35: Original: 0 | Attacked: 1 | RPCA: 0
```

(b) RPCA results of images attacked with BIM

Fig. 4.4 RPCA results images attacked with FGSM and images attacked with BIM

```
Original Labels | Attacked Predictions | RPCA Predictions
Image 1: Original: 3 | Attacked: 5 | RPCA: 7
Image 2: Original: 2 | Attacked: 5 | RPCA: 2
Image 3: Original: 2 | Attacked: 5 | RPCA: 2
Image 4: Original: 4 | Attacked: 0 | RPCA: 0
Image 5: Original: 6 | Attacked: 0 | RPCA: 6
Image 6: Original: 7 | Attacked: 0 | RPCA: 6
Image 7: Original: 7 | Attacked: 0 | RPCA: 0
Image 8: Original: 5 | Attacked: 2 | RPCA: 2
Image 9: Original: 6 | Attacked: 5 | RPCA: 7
Image 10: Original: 4 | Attacked: 5 | RPCA: 7
Image 11: Original: 4 | Attacked: 1 | RPCA: 7
Image 12: Original: 6 | Attacked: 0 | RPCA: 6
Image 13: Original: 6 | Attacked: 0 | RPCA: 6
Image 14: Original: 1 | Attacked: 5 | RPCA: 0
Image 15: Original: 1 | Attacked: 0 | RPCA: 6
Image 16: Original: 4 | Attacked: 4 | RPCA: 4
Image 17: Original: 7 | Attacked: 5 | RPCA: 7
Image 18: Original: 6 | Attacked: 0 | RPCA: 6
Image 19: Original: 4 | Attacked: 5 | RPCA: 0
Image 20: Original: 0 | Attacked: 0 | RPCA: 0
Image 21: Original: 0 | Attacked: 1 | RPCA: 0
Image 22: Original: 0 | Attacked: 5 | RPCA: 0
Image 23: Original: 1 | Attacked: 5 | RPCA: 0
Image 24: Original: 7 | Attacked: 0 | RPCA: 7
Image 25: Original: 0 | Attacked: 1 | RPCA: 6
Image 26: Original: 4 | Attacked: 5 | RPCA: 0
Image 27: Original: 2 | Attacked: 5 | RPCA: 2
Image 28: Original: 2 | Attacked: 5 | RPCA: 2
Image 29: Original: 5 | Attacked: 2 | RPCA: 2
Image 30: Original: 4 | Attacked: 5 | RPCA: 2
Image 31: Original: 5 | Attacked: 2 | RPCA: 2
Image 32: Original: 1 | Attacked: 5 | RPCA: 1
Image 33: Original: 7 | Attacked: 0 | RPCA: 7
Image 34: Original: 3 | Attacked: 5 | RPCA: 7
Image 35: Original: 0 | Attacked: 5 | RPCA: 0
```

(a) RPCA results of images attacked with Deep Fool

```
Original Labels | Attacked Predictions | RPCA Predictions
Image 1: Original: 3 | Attacked: 0 | RPCA: 7
Image 2: Original: 2 | Attacked: 5 | RPCA: 2
Image 3: Original: 2 | Attacked: 5 | RPCA: 2
Image 4: Original: 4 | Attacked: 0 | RPCA: 0
Image 5: Original: 6 | Attacked: 0 | RPCA: 6
Image 6: Original: 7 | Attacked: 0 | RPCA: 6
Image 7: Original: 7 | Attacked: 1 | RPCA: 0
Image 8: Original: 5 | Attacked: 2 | RPCA: 2
Image 9: Original: 6 | Attacked: 5 | RPCA: 7
Image 10: Original: 4 | Attacked: 4 | RPCA: 7
Image 11: Original: 4 | Attacked: 4 | RPCA: 7
Image 12: Original: 6 | Attacked: 0 | RPCA: 6
Image 13: Original: 6 | Attacked: 0 | RPCA: 6
Image 14: Original: 1 | Attacked: 5 | RPCA: 0
Image 15: Original: 1 | Attacked: 5 | RPCA: 6
Image 16: Original: 4 | Attacked: 4 | RPCA: 4
Image 17: Original: 7 | Attacked: 5 | RPCA: 7
Image 18: Original: 6 | Attacked: 0 | RPCA: 6
Image 19: Original: 4 | Attacked: 5 | RPCA: 2
Image 20: Original: 0 | Attacked: 0 | RPCA: 0
Image 21: Original: 0 | Attacked: 1 | RPCA: 0
Image 22: Original: 0 | Attacked: 0 | RPCA: 0
Image 23: Original: 1 | Attacked: 1 | RPCA: 0
Image 24: Original: 7 | Attacked: 0 | RPCA: 7
Image 25: Original: 0 | Attacked: 1 | RPCA: 6
Image 26: Original: 4 | Attacked: 5 | RPCA: 0
Image 27: Original: 2 | Attacked: 2 | RPCA: 2
Image 28: Original: 2 | Attacked: 5 | RPCA: 2
Image 29: Original: 5 | Attacked: 2 | RPCA: 2
Image 30: Original: 4 | Attacked: 4 | RPCA: 2
Image 31: Original: 5 | Attacked: 5 | RPCA: 2
Image 32: Original: 1 | Attacked: 5 | RPCA: 1
Image 33: Original: 7 | Attacked: 0 | RPCA: 7
Image 34: Original: 3 | Attacked: 5 | RPCA: 7
Image 35: Original: 0 | Attacked: 0 | RPCA: 0
```

(b) RPCA results of images attacked with C&W

Fig. 4.5 RPCA results on images attacked with Deep fool and images attacked with C&W

4.1.3 Frequency-Based Adversarial Attack Results

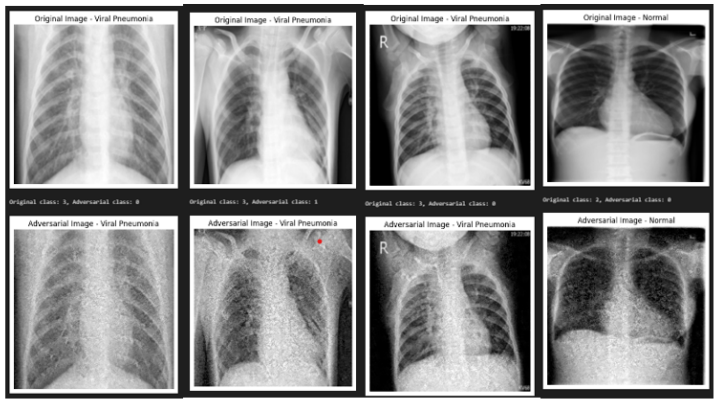

Fig. 4.6 Comparison between original images and frequency based adversarial attacked images from chest X-Ray dataset

Attack Success Rate: 90.00%

	Original Class	Adversarial Class
0	2	0
1	2	0
2	2	0
3	2	0
4	0	1
5	0	0
6	2	0
7	2	0
8	2	2
9	2	0
10	1	0
11	0	1
12	0	0
13	1	0
14	1	0
15	1	0
16	1	0
17	2	1
18	1	2
19	1	0
20	3	0
21	3	3
22	3	1
23	3	0
24	3	0
25	3	0
26	3	0
27	3	0
28	3	0
29	3	0
30	0	1
31	3	0
32	0	3
33	0	1
34	0	1
35	0	1
36	0	1
37	0	1
38	0	3
39	0	1

Fig. 4.7 Attack results on Chest X-Ray images using frequency based attack with an attack success rate of 90%

Chapter 5

Work plan

5.1 Ongoing Work

- **RPCA Implementation and Testing:** Continue implementing the RPCA mechanism on frequency-based adversarial attacks. Further testing across multiple datasets will help refine our method's effectiveness.
- **Fine-Tuning:** Fine-tune the RPCA parameters to optimize defense performance, ensuring the system remains accurate and reliable under diverse adversarial conditions.
- **Performance Evaluation:** Test the effectiveness of RPCA by comparing the accuracy of defended images with that of original, unperturbed images, focusing on diagnostic consistency.

5.2 Future Work

- **Development of Robust Defense Mechanisms:** Investigate and develop more advanced and resilient defense techniques that can detect and counter complex, frequency-based adversarial attacks.

- **Exploration of Novel Adversarial Attack Types:** Research new forms of adversarial attacks that our current defense mechanism cannot easily detect, further challenging our model's robustness.
- **Real-World Application Testing:** Plan for testing in real-world scenarios to validate the defense mechanism's reliability in clinical settings, with a focus on maintaining high diagnostic accuracy and security.

Chapter 6

Conclusion

In this project, we explored and implemented a defense mechanism against frequency-based adversarial attacks in medical image diagnosis using a ResNet model and RPCA-based high-frequency component filtering. Our results indicate that RPCA is effective in filtering adversarial perturbations while preserving critical diagnostic information. Testing with various attack types showed enhanced model robustness, confirming the viability of our approach in securing deep learning-driven diagnostic systems.

This research contributes to the field by establishing a foundation for future advancements in defense mechanisms, specifically against complex, frequency-targeted adversarial attacks in medical imaging. By improving model resilience, this project supports the development of reliable AI-driven healthcare diagnostics.

References

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

[2] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

[3] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. R. Roth, and D. Xu, "Unetr: Transformers for 3d medical image segmentation," in *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022, pp. 1748–1758.

[4] G. Bortsova, C. González-Gonzalo, S. C. Wetstein, F. Dubost, I. Katramados, L. Hogeweg, B. Liefers, B. van Ginneken, J. P. Pluim, M. Veta, C. I. Sánchez, and M. de Bruijne, "Adversarial attack vulnerability of medical image analysis systems: Unexplored factors," *Medical Image Analysis*, vol. 73, p. 102141, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841521001870

[5] Y. Guo, Y. Liu, T. Georgiou, and M. Lew, "A review of semantic segmentation using deep neural networks," *International Journal of Multimedia Information Retrieval*, vol. 7, 06 2018.

[6] F. Chen, J. Wang, H. Liu, W. Kong, Z. Zhao, L. Ma, H. Liao, and D. Zhang, "Frequency constraint-based adversarial attack on deep neural networks for medical image classification," *Computers in Biology and Medicine*, vol. 164, p. 107248, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0010482523007138

[7] Z.-Q. Zhao, P. Zheng, S. tao Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 3212–3232, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:49862415

[8] T. Shabeera, S. M. Kumar, and P. Chandran, "Curtailing job completion time in mapreduce clouds through improved virtual machine allocation," *Computers & Electrical Engineering*, vol. 58, pp. 190 – 202, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0045790616305286

[9] X. Ma, Y. Niu, L. Gu, Y. Wang, Y. Zhao, J. Bailey, and F. Lu, "Understanding adversarial attacks on deep learning based medical image analysis systems," *Pattern Recognition*, vol. 110, p. 107332, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320320301357

[10] X. Wei, R. Yu, and J. Sun, "View-gcn: View-based graph convolutional network for 3d shape analysis," 06 2020, pp. 1847–1856.

[11] M. P. Gilesh, S. D. M. Kumar, and L. Jacob, "HyViDE: A Framework for Virtual Data Center Network Embedding," in *Proceedings of the Symposium on Applied Computing*, ser. SAC '17. New York, NY, USA: ACM, 2017, pp. 378–383. [Online]. Available: http://doi.acm.org/10.1145/3019612.3019629

National Institute of Technology Calicut

Department of Electronics and Communication Engineering

B. Tech Project: Mid-Sem Evaluation

Continuous Evaluation Form

Project Title: Adversial Attacks and Defense strategies for Improving Accuracy in Medical Image Diagnosis

Group No. GS12

Guide: Dr. Ameer P. M.

	Roll No. (Eg. B21xxxxx1EC) B210742EC	**Roll No.** (Eg. B21xxxxx2EC) B210708EC	**Roll No.** (Eg. B21xxxxx3EC) B210658EC	**Roll No.** (Eg. B21xxxxx4EC) B210612EC	**Roll No.** (Eg. B21xxxxx5EC) B210761EC
	Student Name 1 M. Ajay Kumar Naidu	*Student Name 2* Mutyala Veera Abhinanda	*Student Name 3* Nethala Vijay Praneeth	*Student Name 4* Pranav Kandukuru	*Student Name 5* Liyana N K
Meeting 1 *(5/8/2024 to 12/8/2024)*	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓
	Guide's Comments, if any: Progress of project is good				Guide signature with date: 9/8/24
Meeting 2 *(13/8/2024 to 20/8/2024)*	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓
	Guide's Comments, if any				Guide signature with date: 13/8/24
Meeting 3 *(21/8/2024 to 25/8/2024)*	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓
	Guide's Comments, if any				Guide signature with date: 21/8/24
Meeting 4 *(26/8/2024 to 1/9/2024)*	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓	Present/Absent ✓
	Guide's Comments, if any				Guide signature with date: 26/8/24

National Institute of Technology Calicut

Department of Electronics and Communication Engineering

B. Tech Project: End-Sem Evaluation

Continuous Evaluation Form

Project Title: Adversial Attacks and Defence strategies for Improving Accuracy in Medical Image Diagnosis

Group No. GS12 Guide: Dr. Ameer PM

	Roll No. (Eg. B21xxxxx1EC) B210742EC	**Roll No.** (Eg. B21xxxxx2EC) B210708EC	**Roll No.** (Eg. B21xxxxx3EC) B210658EC	**Roll No.** (Eg. B21xxxxx4EC) B210612EC	**Roll No.** (Eg. B21xxxxx5EC) B210761EC
	Student Name 1 M Ajay Kumar Naidu	*Student Name 2* Muttyala Veera Abhinash	*Student Name 3* N Vijay Praneeth	*Student Name 4* Pranav kandukuru	*Student Name 5* Liyana NK
Meeting 5 *(1/10/2024 to 8/10/2024)*	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~
	Guide's Comments, if any				Guide signature with date 8/10/24
Meeting 6 *(9/10/2024 to 15/10/2024)*	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~
	Guide's Comments, if any				Guide signature with date 15/10/24
Meeting 7 *(16/10/2024 to 22/10/2024)*	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~
	Guide's Comments, if any				Guide signature with date 22/10/24

Meeting 8 *(23/10/2024 to 30/10/2024)*	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~	Present/~~Absent~~
	Guide's Comments, if any				Guide signature with date: 30/10/24