

# Introduction to Deep Learning Group Project Report

Study the Swin Transformer for

Medical Image Classification

**Members:**

**BI13 - 093 Nguyễn Ngọc Anh Đức**

**BI13 - 242 Nguyễn Hoàng Lân**

**BI13 - 022 Nguyễn Hoàng Anh**

**BI13 - 163 Trương Xuân Hiếu**

**BI13 - 338 Lê Phúc Nguyên**

**BI13 - 018 Lê Nhật Anh**

**Table of Contents**

[**I. Introduction 1**](#_gjdgxs)

[1. Overview 1](#_30j0zll)

[2. The Shifted Windows Transformer (Swin) 1](#_1fob9te)

[**II. Methodology and Materials 3**](#_3znysh7)

[1. Overall methods: 3](#_2et92p0)

[2. Dataset 3](#_tyjcwt)

[3. Model 4](#_3dy6vkm)

[3.1. Architecture 4](#_750pcv7nb1cw)

[3.2. Two Successive Swin Transformer Blocks 5](#_eel05zodasyo)

[**III. Implementation 7**](#_1t3h5sf)

[1. Model Structure and Training Setting 7](#_q0s692lgy21f)

[2. Optimizer and Loss Function: 7](#_yxph4pl43y81)

[3. Training and Results 7](#_js6nc15lf6z5)

[b. For result: 8](#_6xiwpoz7b0rr)

[c. Confusion Matrix: 9](#_io1k6whg61a2)

[**IV. Conclusion: 10**](#_4i7ojhp)

[1. What we have learned 10](#_2xcytpi)

[2. What to expect in the future 10](#_1ci93xb)

**Abstract:**

The Transformer model has become the dominant architecture in natural language processing (NLP), powering advanced models like GPT and being applied to tasks such as keyword recognition and voice detection. However, its application in computer vision remains relatively limited. Currently, convolutional neural networks (CNNs) still lead the field in image classification, with models like AlexNet, ResNet, and VGGNet at the forefront. Traditionally, Transformers in vision tasks have been combined with convolutional layers or used to replace specific components. Recent research shows that such hybrid approaches are unnecessary, as pure Transformer models applied to sequences of image patches can achieve strong performance on image classification tasks. The Swin Transformer, in particular, has shown impressive results compared to state-of-the-art CNNs, offering strong performance on image classification tasks. It also proves more efficient in terms of computational resources, especially when pre-trained on large datasets and fine-tuned on mid-sized or smaller image recognition benchmarks. Its hierarchical architecture and ability to handle multi-scale features make it highly effective for various vision tasks beyond classification, such as object detection and segmentation.

# Introduction

## Overview

This report aims to study and analyze the Swin Transformer. The focus will be on understanding the overall architecture, its functionality, and its advantages over traditional convolutional neural networks (CNNs). The report will provide readers with insights into the key aspects of the Swin Transformer and its significance in advancing computer vision tasks.

## The Shifted Windows Transformer (Swin)

The Swin Transformer represents a significant evolution in the application of Transformer models to computer vision. Unlike CNNs, which process entire images through convolutional layers, Swin introduces a more efficient and scalable approach by using shifted windows. This hierarchical structure enables the model to capture both local and global features while reducing computational complexity. Instead of processing an image as a whole, Swin Transformer divides the image into smaller patches (or windows), similar to how ViT handles image patches. However, Swin’s innovation lies in the way it shifts the windows at each layer, allowing for better cross-window connections and multi-scale feature learning.

Swin Transformer excels in a variety of vision tasks, including image classification, object detection, and semantic segmentation. It has demonstrated state-of-the-art performance across several benchmarks, often surpassing CNN-based models like ResNet. The Swin Transformer’s hierarchical nature and ability to handle multi-scale features make it particularly efficient, scaling well across different image sizes and complexities, while requiring fewer computational resources compared to previous Transformer-based mod

# Methodology and Materials

## Overall methods:

The Swin Transformer, designed for computer vision tasks, is highly effective for image classification, object detection, and segmentation. Since it’s structured to handle large-scale datasets efficiently, we utilized a pre-trained Swin Transformer model. Pre-trained models allow us to leverage the learning from large datasets, reducing computational cost and training time.

We conducted four different experiments. In the first experiment, we built a smaller-scale model based on the Swin Transformer architecture from scratch. The second experiment used a PyTorch implementation of the Swin Transformer without pre-training to train the model on our dataset. In the third experiment, we fine-tuned a pre-trained Swin Transformer model, pre-trained on ImageNet21k, using our target dataset. Finally, for the last experiment, we implemented a fine-tuning process found in literature to delve deeper into Swin Transformer's performance on different datasets.

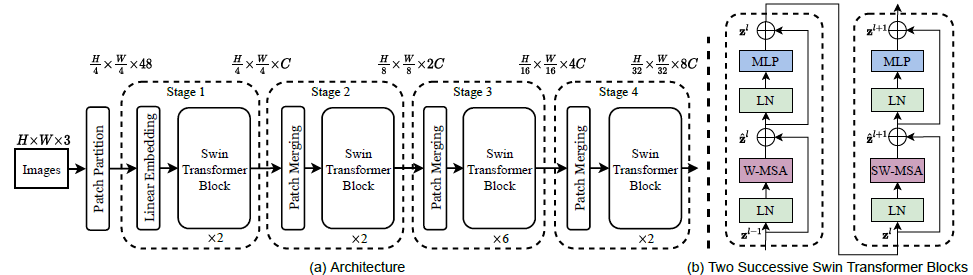
## Dataset

Our model is trained on the ChestXray14\_v3 dataset. The provided dataset is structured into three main directories: train, val and test. These directories represent different subsets of the data for training, validation, and testing, respectively. Each subset contains three separate categories:

* Normal: This folder includes medical images of healthy individuals with no signs of pneumonia. (train: 7081 images; val: 885 images; test: 885 images)
* Pneumonia: This folder consists of images indicating pneumonia cases, which can help differentiate between pneumonia and other conditions. (train: 4836 images; val: 605 images; test: 604 images)

## Model

### Architecture



This image illustrates the overall architecture of the Swin Transformer, which processes an image through different stages.

1. Patch Partition:

* Input images of size H×W×3H (with height H, width W, and 3 color channels for RGB) are divided into non-overlapping patches.
* Each patch is treated as a token, similar to how words are treated in NLP models. For example, an image is divided into smaller​ x patches.

1. Linear Embedding:

* After partitioning, each patch is linearly embedded into a vector of size 48. This embedding projects the raw pixel values of the patches into a higher-dimensional space.

1. Stages:

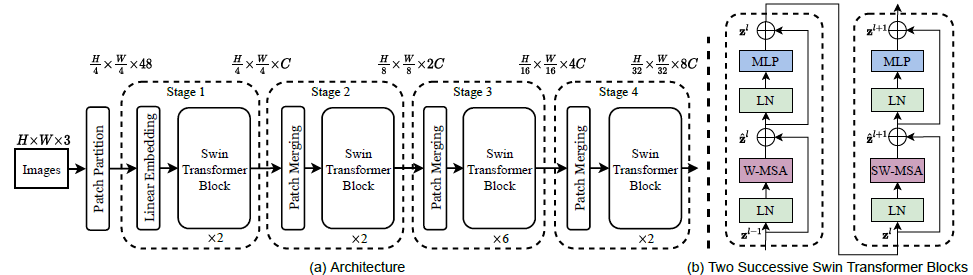
The architecture has 4 stages, with each stage containing Swin Transformer blocks. As we move from one stage to the next, the resolution of the feature map decreases, but the number of channels increases, allowing the network to capture higher-level features.

* Stage 1:
  + Input size: x x 48
  + Contains 2 successive Swin Transformer blocks (as shown in section (b)).
* Stage 2:
  + After applying Patch Merging, the spatial size of the feature map is reduced to x x 48, and the channel dimension is doubled to CCC.
  + The stage contains 2 Swin Transformer blocks.
* Stage 3:
  + After another patch merging step, the spatial resolution becomes x ​, and the channel dimension increases to 2C.
  + This stage contains 6 Swin Transformer blocks.
* Stage 4:
  + After the final patch merging, the spatial size reduces to x , and the number of channels is now 8C.
  + This stage contains 2 Swin Transformer blocks.

1. Patch Merging

* At the end of each stage (except the last), the feature maps undergo a process called Patch Merging.
* Patch merging combines neighboring patches (i.e., reduces spatial dimensions) and increases the number of channels (by concatenating them), allowing the model to capture more abstract features as the resolution decreases.

### Two Successive Swin Transformer Blocks



On this image, you can see two Swin Transformer blocks, which are the building blocks used in each stage.

1. Layer Normalization (LN):

* Each block starts with a Layer Normalization step, applied before both the **Multi-Head Self-Attention** (MSA) and **MLP** components.

1. Window-based Multi-Head Self-Attention (W-MSA):

* In a traditional Vision Transformer, self-attention would be applied globally across the entire image. In contrast, the Swin Transformer applies self-attention within **local windows** of the image.
* These windows are smaller regions of the image, improving efficiency by reducing the computational complexity of self-attention.

1. Shifted Window Multi-Head Self-Attention (SW-MSA):

* To introduce interactions between different windows (without making the computation too expensive), the Swin Transformer applies **Shifted Window Attention**.
* In this case, the windows are shifted by a fixed amount, so the next block can compute attention on regions that overlap with the previous block, allowing for cross-window communication.

1. MLP:

* After attention, the output is passed through a **Multi-Layer Perceptron (MLP)**, which consists of two fully connected layers with a GELU nonlinearity in between.

1. Residual Connections:

* Each block includes **residual connections** (skip connections), meaning that the input to each block is added back to its output (after applying attention or MLP). This helps in training deeper networks by mitigating the vanishing gradient problem.

1. Repeated Structure:

* Two successive blocks are shown to illustrate that the first block uses **Window-based MSA**, while the second block uses **Shifted Window MSA**, making this pattern a fundamental part of the Swin Transformer.

# Implementation

## Model Structure and Training Setting

For the initial model implementation, we used a Swin Transformer, a vision transformer-based architecture, designed for chest X-ray classification. The input to this model consists of chest X-ray images, which were resized to 224x224 pixels before being passed into the model. This preprocessing ensures that the images are compatible with the Swin Transformer architecture.

The Swin Transformer uses a hierarchical structure that splits the image into smaller patches and processes them using multi-head self-attention within local windows. This is beneficial for extracting features in medical imaging, as it allows the model to focus on both local and global details of the X-ray images.

The backbone of the Swin Transformer is pre-trained on ImageNet (using IMAGENET1K\_V1 weights). However, to adapt the model for our specific task of chest X-ray classification (with classes "normal" and "pneumonia"), the fully connected classification layer was replaced. We modified the final layer to output two classes by adding a new linear layer with 2 output units.

To fine-tune the model, we chose to freeze the parameters of the pre-trained layers to prevent overfitting, as our dataset contains a smaller number of training samples (compared to large-scale datasets like ImageNet). The final fully connected layer was left trainable to learn the specific features from the chest X-ray dataset.

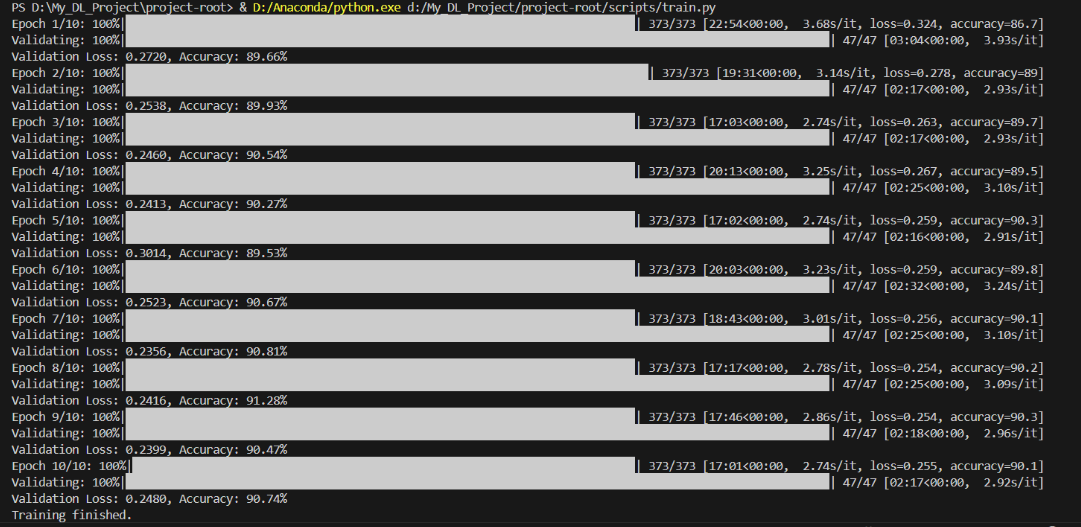
## Optimizer and Loss Function:

* The model was trained using the Adam optimizer with a learning rate of 0.001.
* The loss function employed was CrossEntropyLoss.
* Additionally, weight decay was used to regularize the model and prevent overfitting.

#### 

## Training and Results

* 1. For training, we used the following setup:

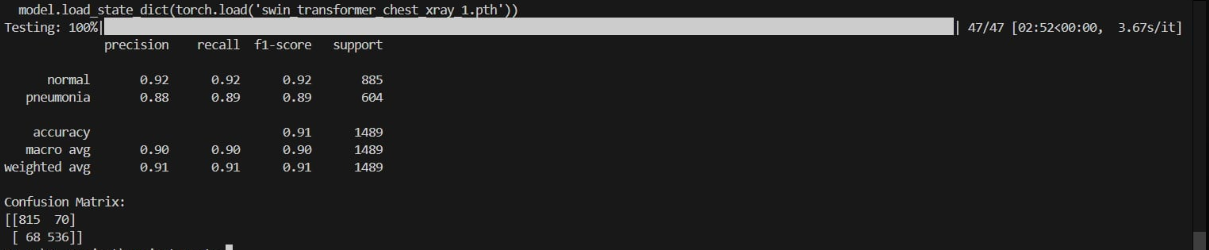


* Batch Size: 32
* Epochs: 10
* Optimizer: Adam
* Learning Rate: 0.001
* Device: GPU (either CUDA if available or CPU)

We divided the dataset into training and validation sets, using standard transforms (resizing, random flipping, and normalization) to augment and prepare the images.

### For result:

After training the model, we evaluated its performance on the test set. Below are the key metrics from the classification report and confusion matrix:



* Accuracy: The model achieved an overall accuracy of 91**%** on the test data.
* Precision:
  + For the "normal" class: 0.92
  + For the "pneumonia" class: 0.88
* Recall:
  + For the "normal" class: 0.92
  + For the "pneumonia" class: 0.89
* F1-score:
  + For the "normal" class: 0.92
  + For the "pneumonia" class: 0.89

The overall weighted averages for precision, recall, and F1-score were approximately **0.91**, indicating the model's balanced performance across both classes.

### Confusion Matrix:

* True Positives:
  + 815 "normal" cases correctly classified.
  + 536 "pneumonia" cases correctly classified.
* False Positives:
  + 70 normal cases incorrectly classified as pneumonia.
  + 68 pneumonia cases incorrectly classified as normal.

These results show that the model performs well in distinguishing between "normal" and "pneumonia" cases, with slight misclassifications between the two.

# Conclusion:

## What we have learned

Through our study of the **Swin Transformer** for medical image classification, we have gained valuable insights into how this model can effectively process medical images by leveraging its hierarchical architecture.

By dividing images into patches, applying self-attention within local windows, and progressively merging patches, the Swin Transformer captures both local and global context, which is critical for detecting subtle features in medical images such as tumors or lesions. The use of shifted windows further improves the model’s ability to share information across different parts of the image, ensuring that both fine-grained details and larger anatomical structures are captured efficiently.

Additionally, the architecture’s scalability and ability to handle varying image resolutions make it a powerful candidate for medical imaging tasks, where high resolution and precise localization are essential.

## What to expect in the future

In the future, we can expect further advancements in applying Swin Transformers to medical image classification. Research may focus on optimizing the model for specific medical domains, such as radiology, pathology, and ophthalmology, tailoring it to detect diseases with higher accuracy. Hybrid models combining Swin Transformers with convolutional neural networks (CNNs) or incorporating domain-specific priors could also emerge, leveraging the strengths of both architectures.

Additionally, as large-scale annotated medical datasets become more accessible, we could see Swin Transformers used to develop highly robust models that can generalize across diverse clinical scenarios. Future work may also explore integrating interpretability techniques to ensure that model predictions are transparent and trusted by healthcare professionals.