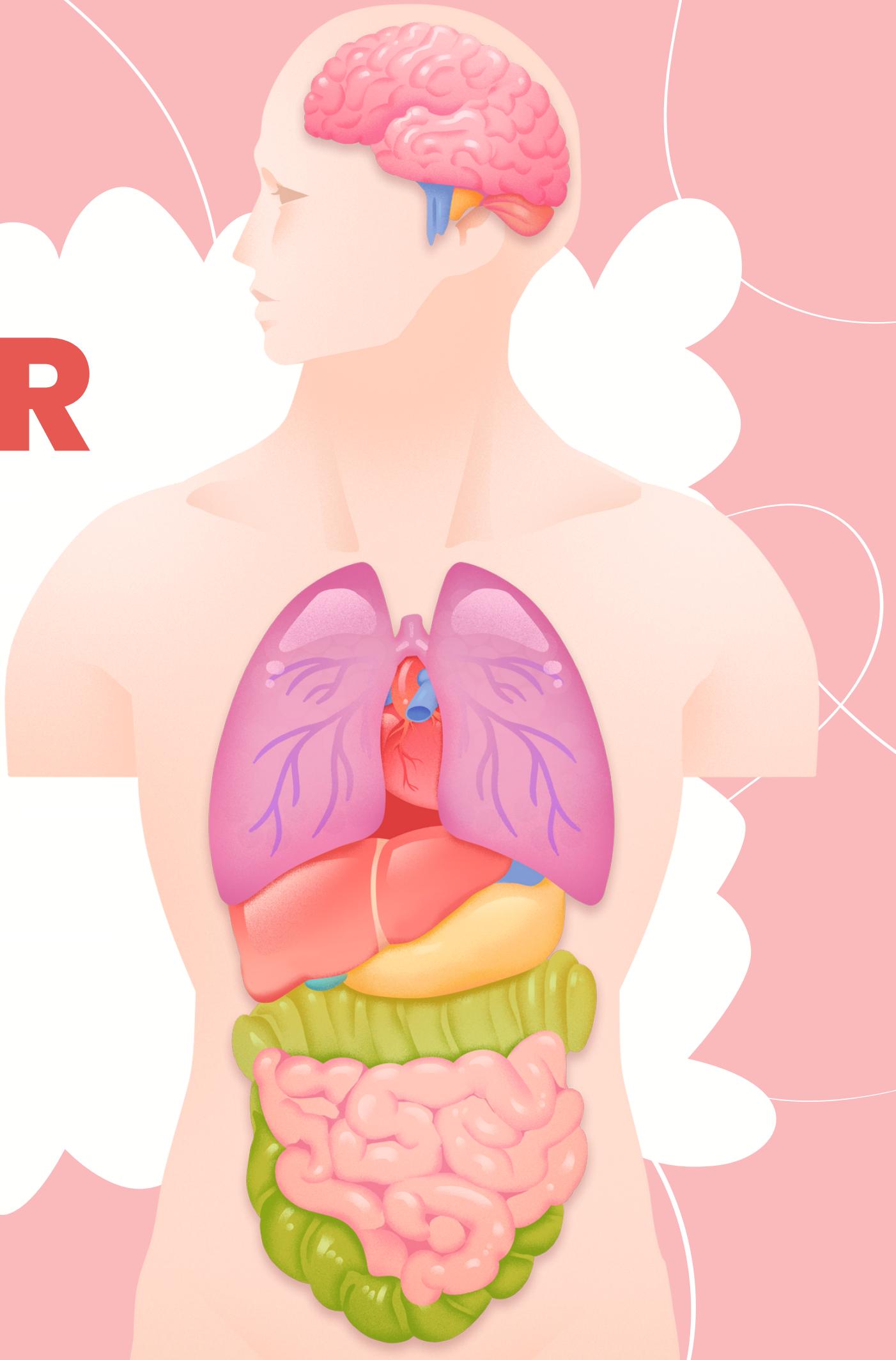


LUNGS CANCER RECURRENCE PREDICTOR



Masters Thesis by



Tathagat Saha

Mat: 902046

University of Milan - Bicocca

Agenda

01

Introduction

02

Problem Statement

03

Dataset

04

Literature
Referred

05

Process Flow

06

Data
Preprocessing

07

Data
Augmentation

08

Dataset
Splitting

09

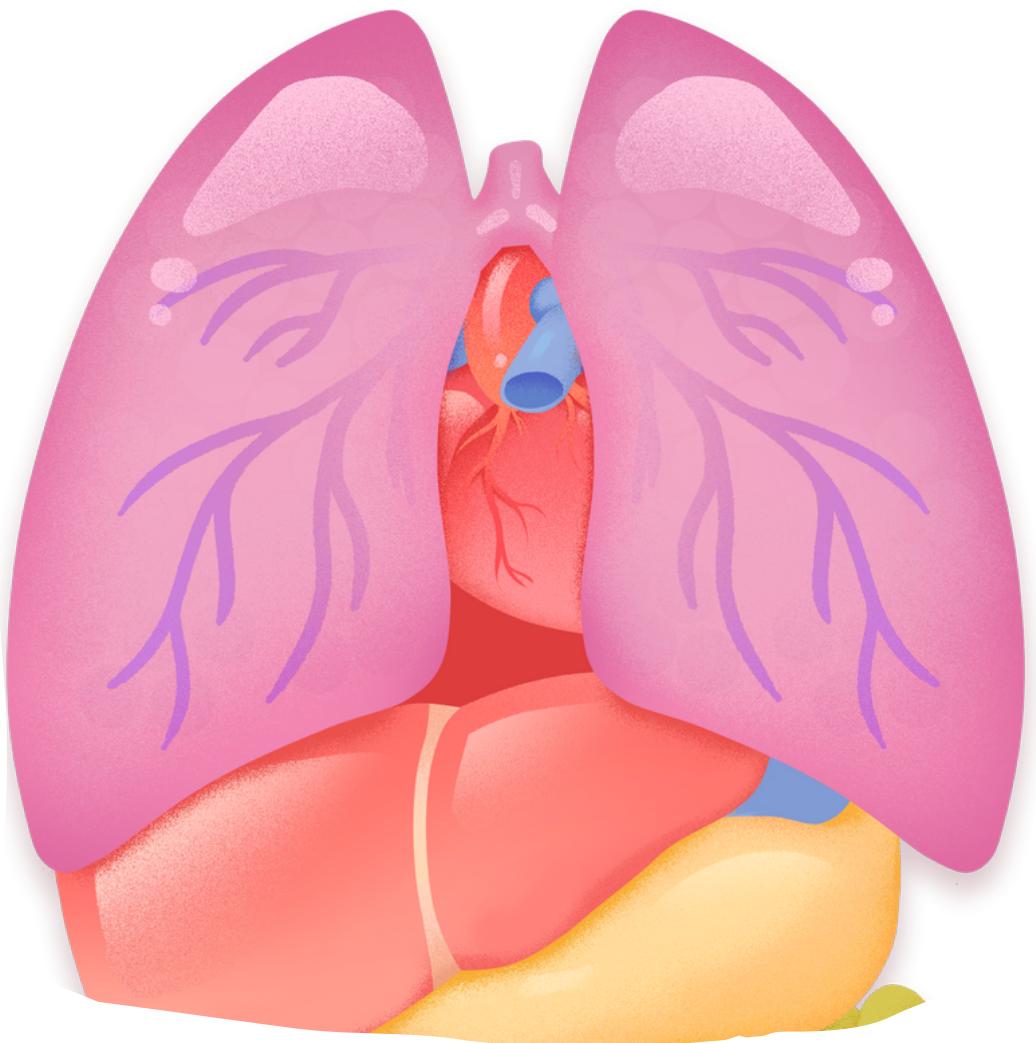
Model Training

10

Performance
Testing

11

Future Steps



Introduction



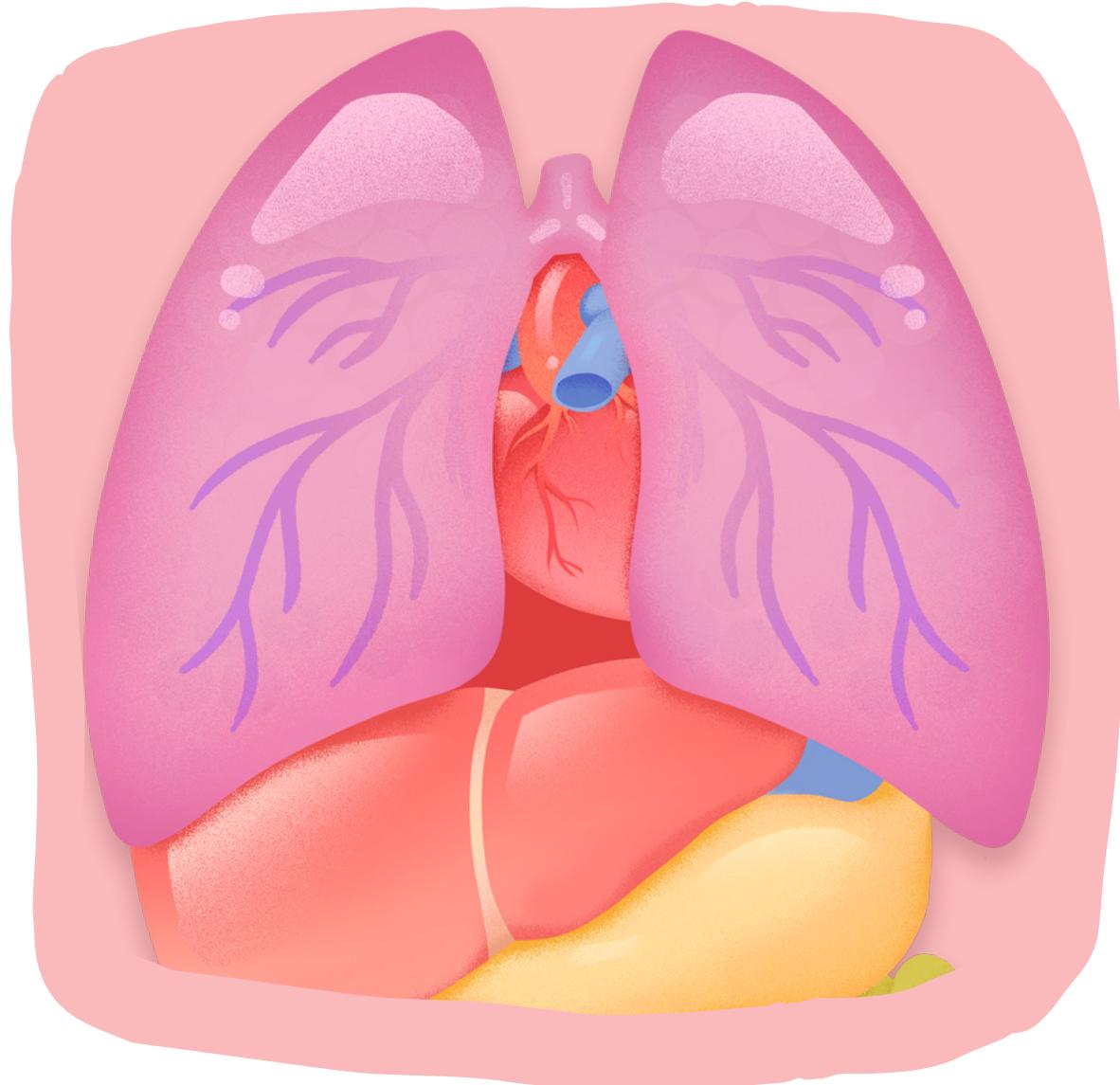
General concepts

A CT scan, or computed tomography scan, is a medical imaging procedure that uses X-rays to create detailed cross-sectional images of the body. It provides more detailed information than conventional X-rays and is commonly used in various medical fields for diagnostic and treatment planning purposes

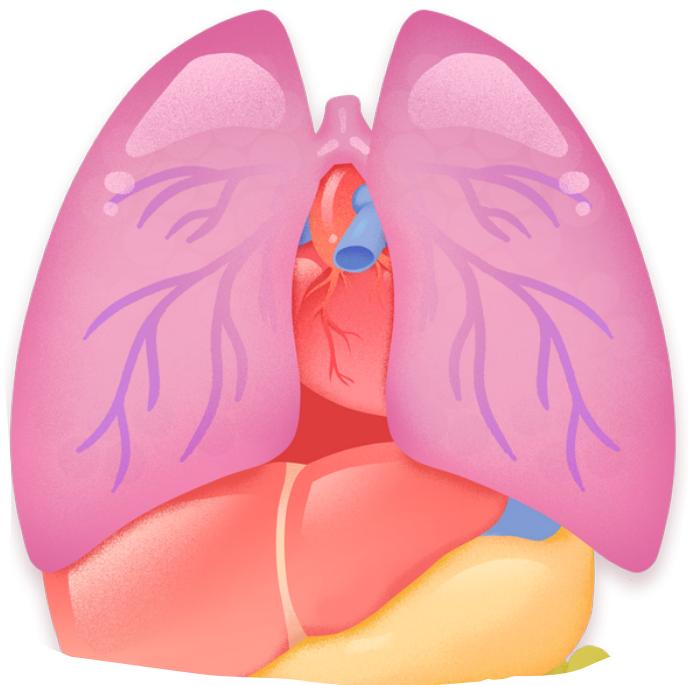


Output

It provides us a 3D view of the affected area in question, by taking multiples 2D slices of the entire area. A combination of all the 2D slices, gives us a holistic 3D view



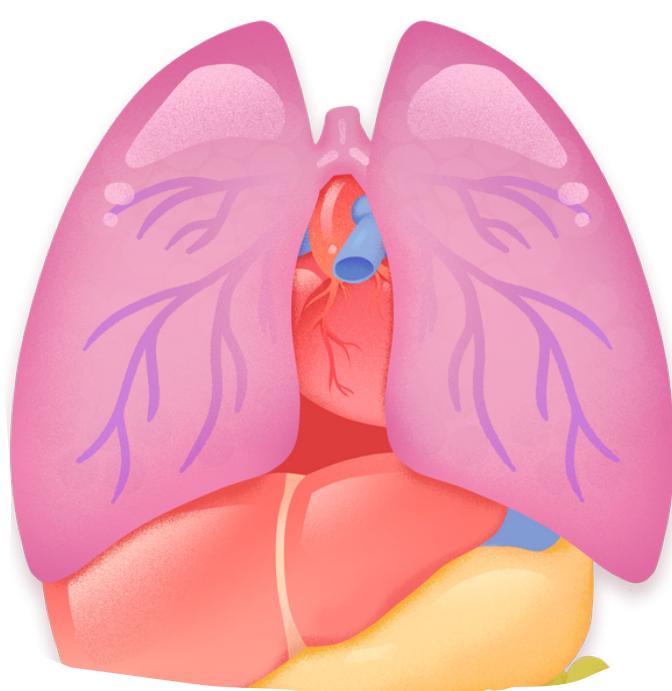
Problem Statement



Prediction of recurrence of lungs cancer before the actual surgery of the lungs tumour takes place.

Thus, helping surgeons to take precautionary measures before the actual surgery.

Dataset



01

The data we are using has been obtained from Cancer Imaging Archive. The dataset contains lungs CT scans of 211 patients

02

There also exists a csv file which contains the labels. The dataset has all patients with lungs cancer, but the labels indicate whether they had a recurrence or not

LITERATURE REFERRED



Uniformizing Techniques to Process CT scans with 3D CNNs for Tuberculosis Prediction by Hasib Zunair, Aimon Rahman, Nabeel Mohammed, Joseph Paul Cohen.

https://github.com/hepxpansion/Tatz_Thesis/blob/main/Research%20Papers/2007.13224.pdf



Automated detection and segmentation of non-small cell lung cancer computed tomography images

https://github.com/hepxpansion/Tatz_Thesis/blob/main/Research%20Papers/s41467-022-30841-3.pdf

Process Flow Diagram

Data Loading

Loading the data into numpy arrays from the available nii.gz files using the nibabel library



Labels Mapping

Labels have been acquired from the labels.csv file and mapped



Database Splitting

The given dataset was split in to 210(train), 64(validate), 28 (test)



Performance Testing

After the model is trained, the unused test data has been used for performance testing evaluation.



Data Pre-Processing

Data preprocessing steps have been implemented on the whole dataset.



Data Augmentation

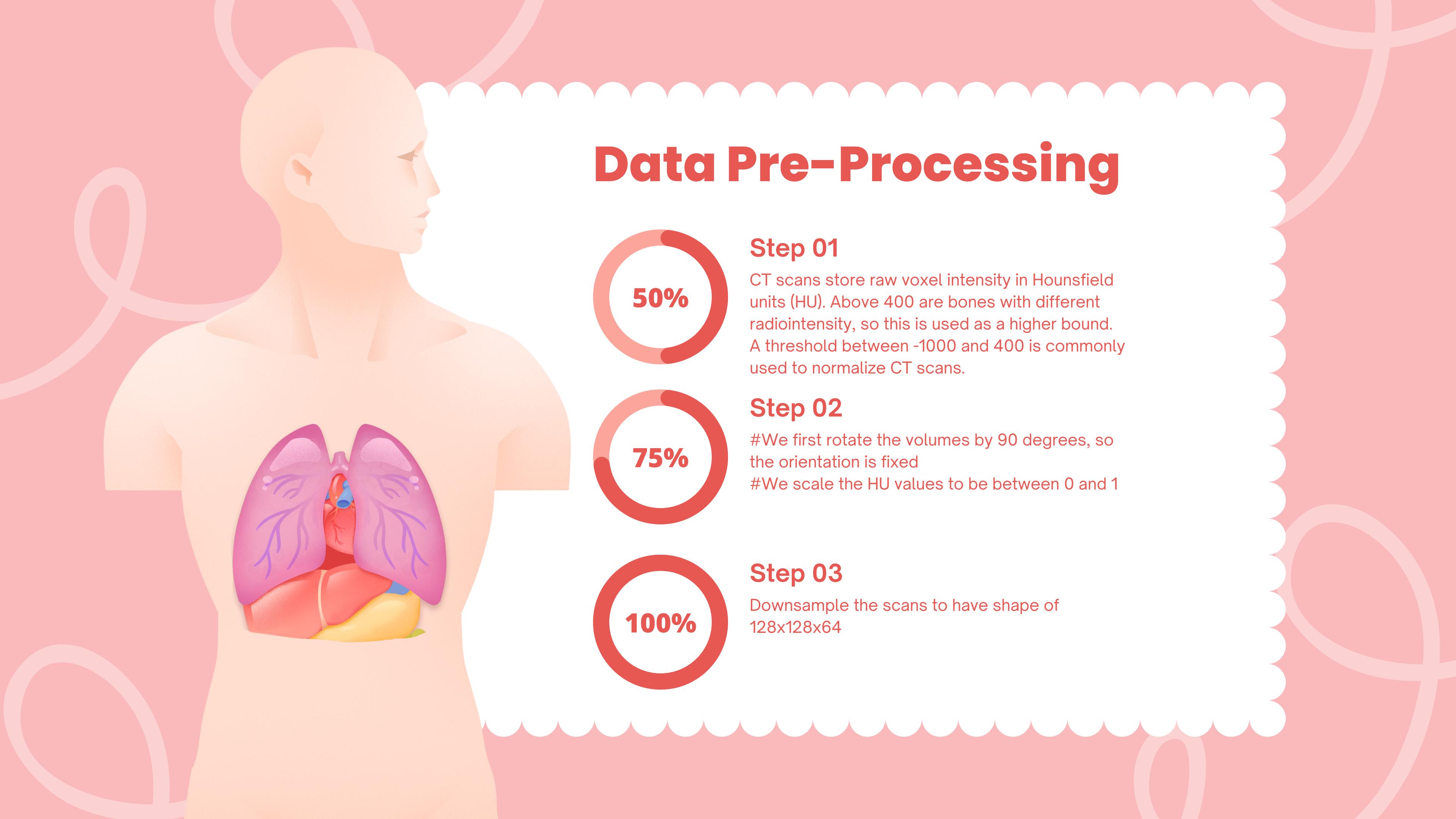
The existing dataset was not balanced, hence Data Augmentation(scaling) was used to balance it.



Model Training

3D CNN has been used as a model and the model was trained with batches of 5





Data Pre-Processing

50%

Step 01

CT scans store raw voxel intensity in Hounsfield units (HU). Above 400 are bones with different radiointensity, so this is used as a higher bound. A threshold between -1000 and 400 is commonly used to normalize CT scans.

75%

Step 02

#We first rotate the volumes by 90 degrees, so the orientation is fixed
#We scale the HU values to be between 0 and 1

100%

Step 03

Downsample the scans to have shape of 128x128x64

BEFORE

CT scans with non-recurrent lung tissue: 156

CT scans with recurrent lung tissue: 54

CONSEQUENCE

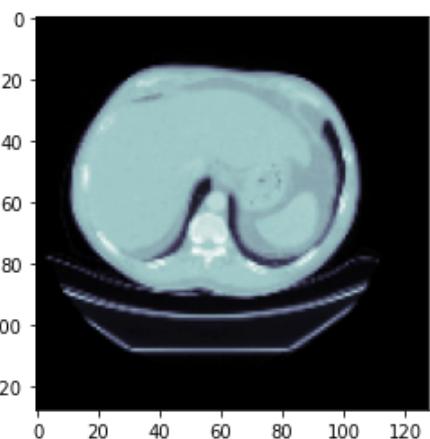
The dataset is not balanced. Training with this dataset will inject a definite bias towards the non-recurrent class as the model will learn more from the more represented class.

AFTER

CT scans with non-recurrent lung tissue: 156

CT scans with recurrent lung tissue: 156

STEPS



Data Augmentation

1. Use Affine transform
2. Apply scaling
3. Resample every image using these transform
4. Convert back to numpy array.
5. Apply the above steps only on the less represented class.

Dataset Splitting

Non-Recurrent = 156
Recurrent = 156

312

Train

220

Rest

92

Validate

64

Test

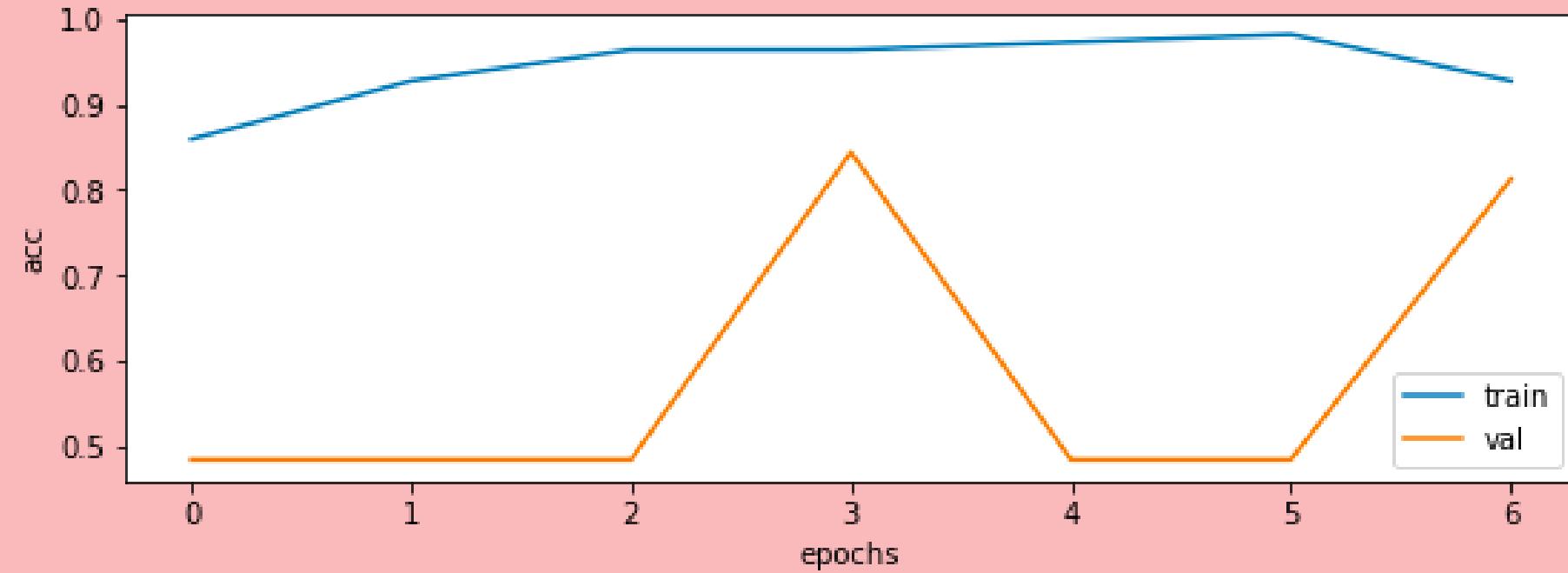
28



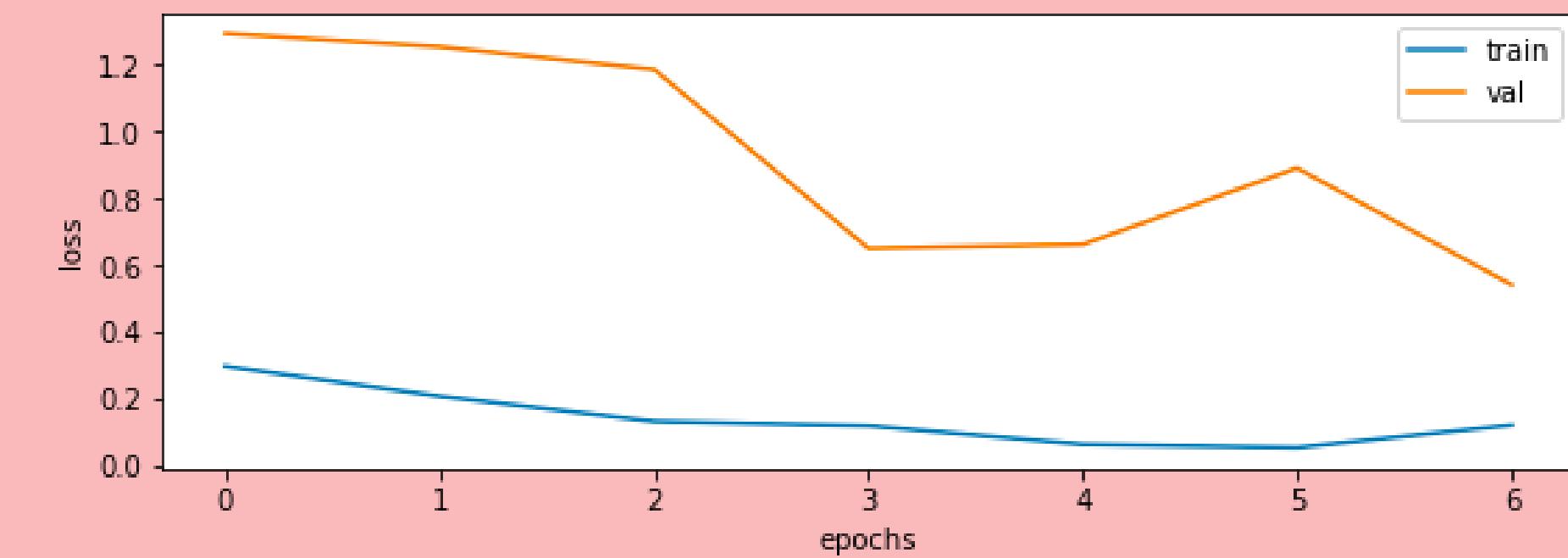
Model Training

Round 1: Training for 7 epochs

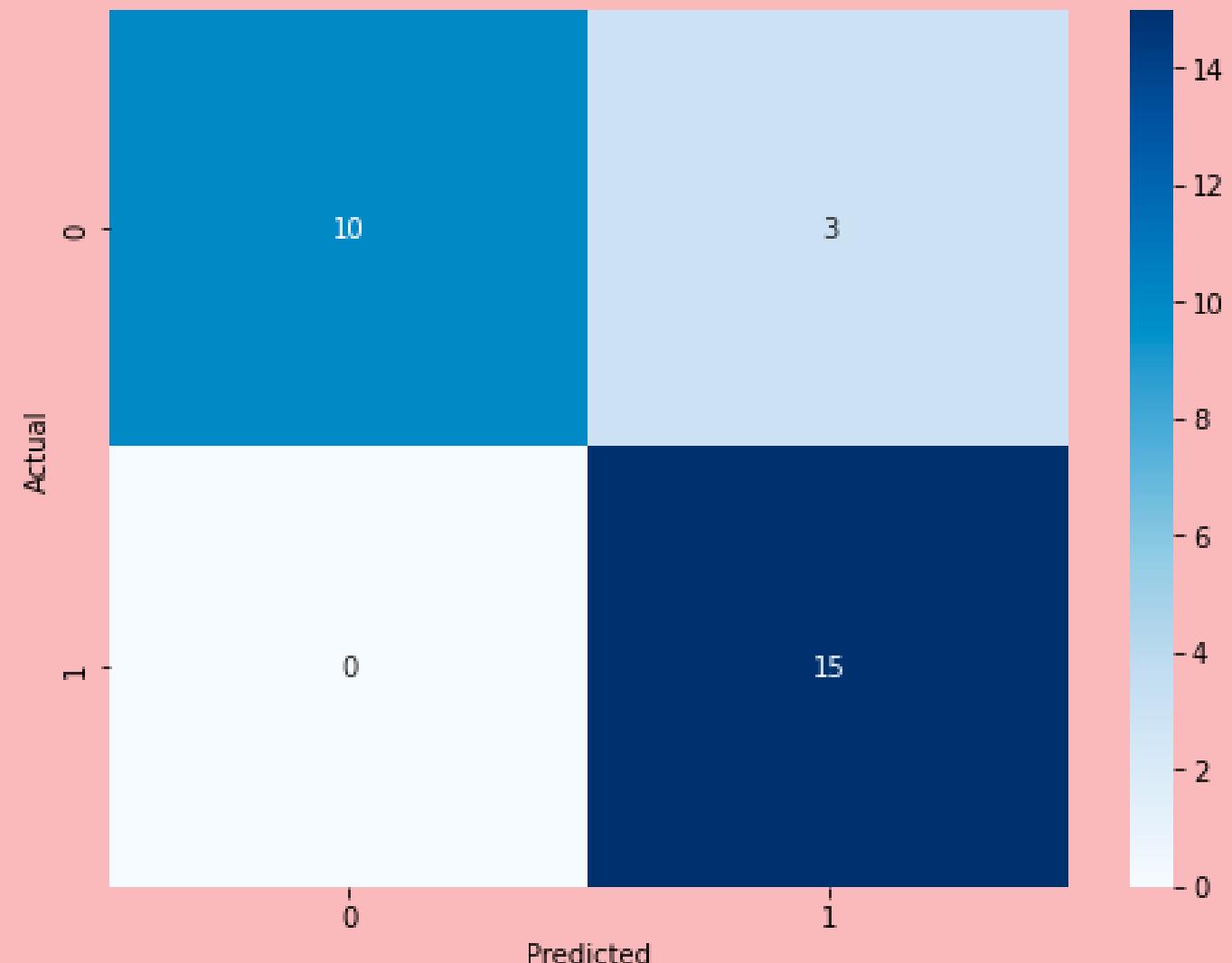
Model acc



Model loss



Confusion Matrix



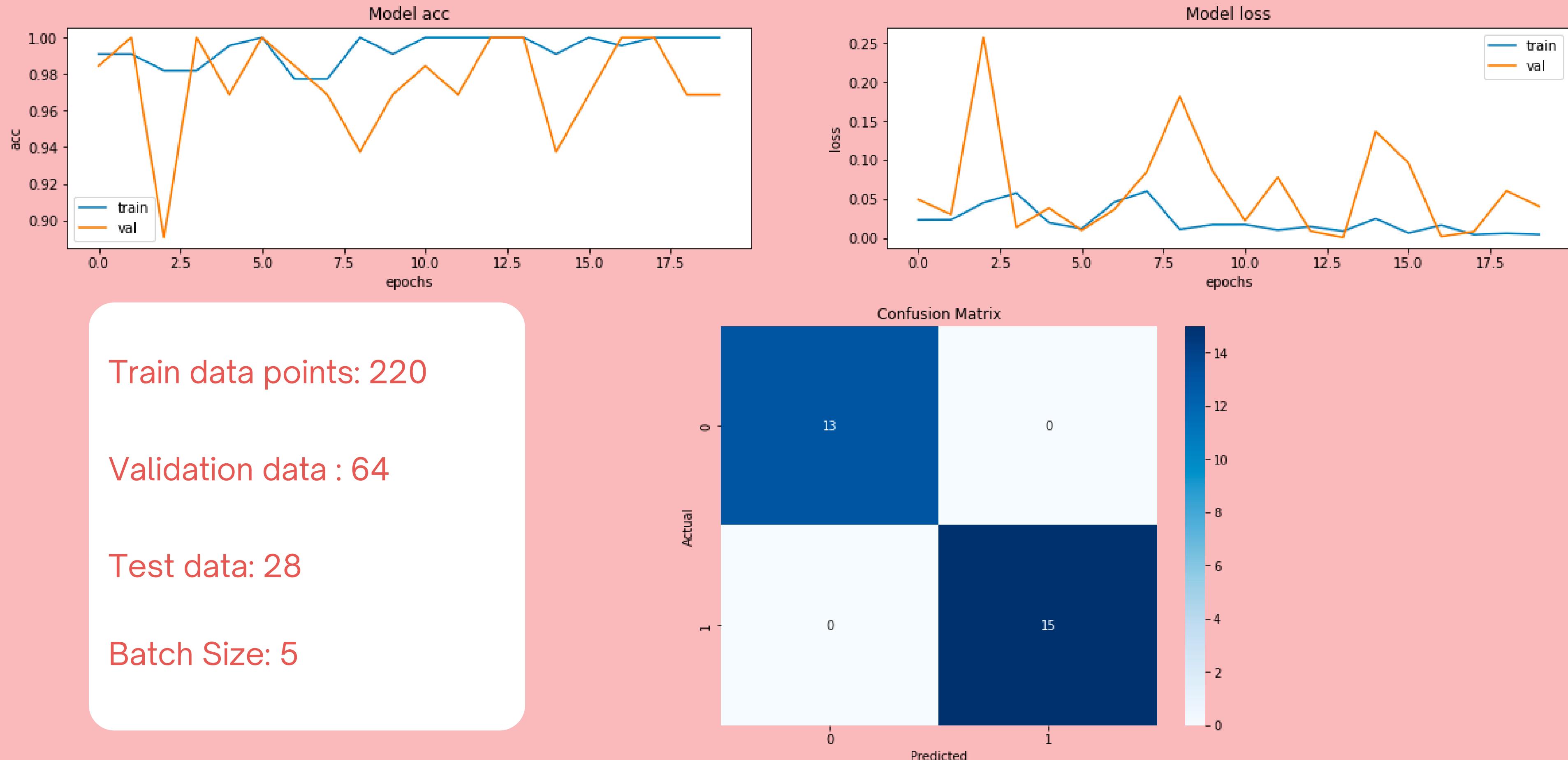
Train data points: 220

Validation data : 64

Test data: 28

Batch Size: 5

Round 2: After training for 20 epochs



Next Steps

Application of 3D GradCam for adding explainability to the model's performance.

**THANK
YOU**