Using FT to enhance the accuracy of ML models in brain tumor detection

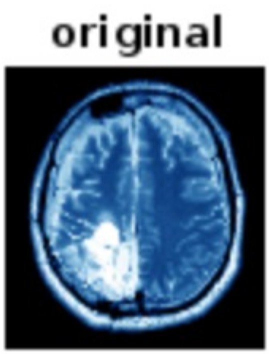
Introduction

The significance in healthcare and medical imaging
enhancing brain tumor detection's accuracy by integrating fourier transform with machine learning
There is a demand for automation in radiology

What is an MRI image

- It’s simply an image that can be decomposed into sin and cosine frequencies
- First change any colored image into gray scale and notice the lower gray levels correspond to a greater amplitude (the brighter area is indicating a tumor)

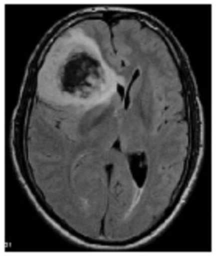
MATLAB



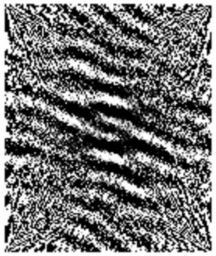
Understanding fourier transform

- FFT2 is a Matlab function that will perform an image Fourier Transform
- The frequency data is projected onto a graph which I happened to label it fourier image
- Each point from this graph corresponds to a particular frequency on the original image

original



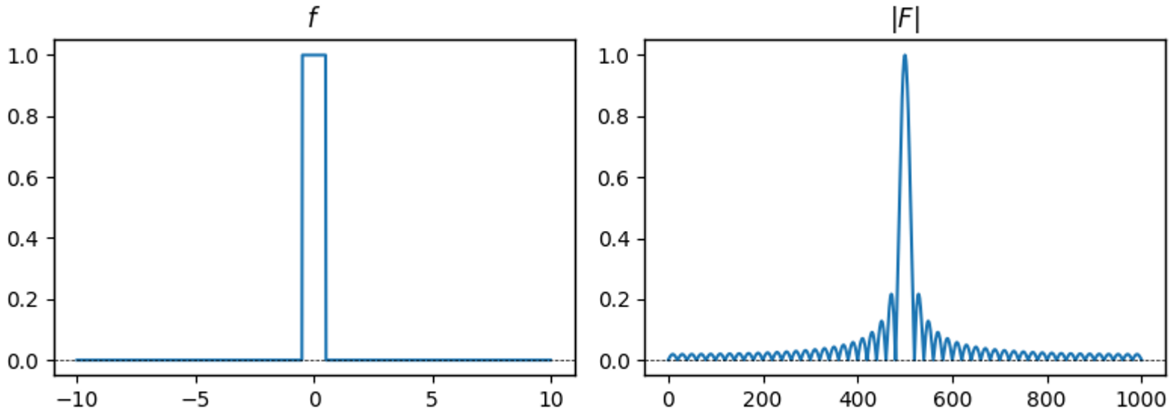
fourier image



FFT

- FT is applied to just 1 row and it goes column by column, think of every column as a space that can be any amplitude

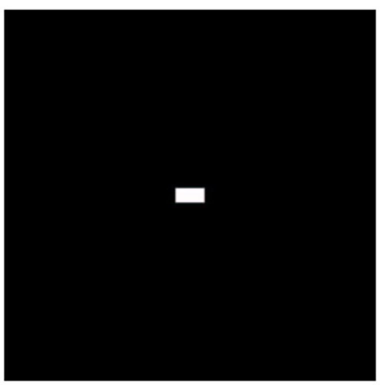
FFT 1D- square wave



FFT2- for 2 dimensions

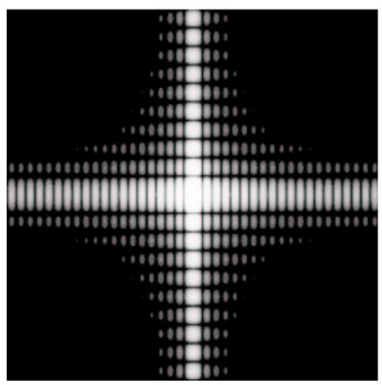
- In the actual image fourier is taken according to shape and brightness of present objects, here the rectangle has the same brightness but everything else is pitch black. Treated as a square wave
- Each row at a time for every column. Meaning at row 1 it goes through every column then it switches to row 2 and goes through every column and so on

An actual image

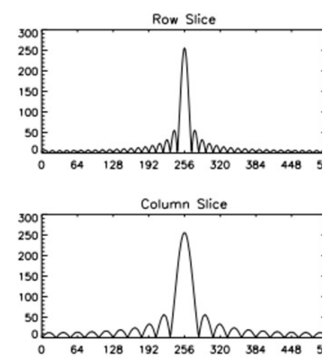


40×20 Rectangle

top view of frequency graph

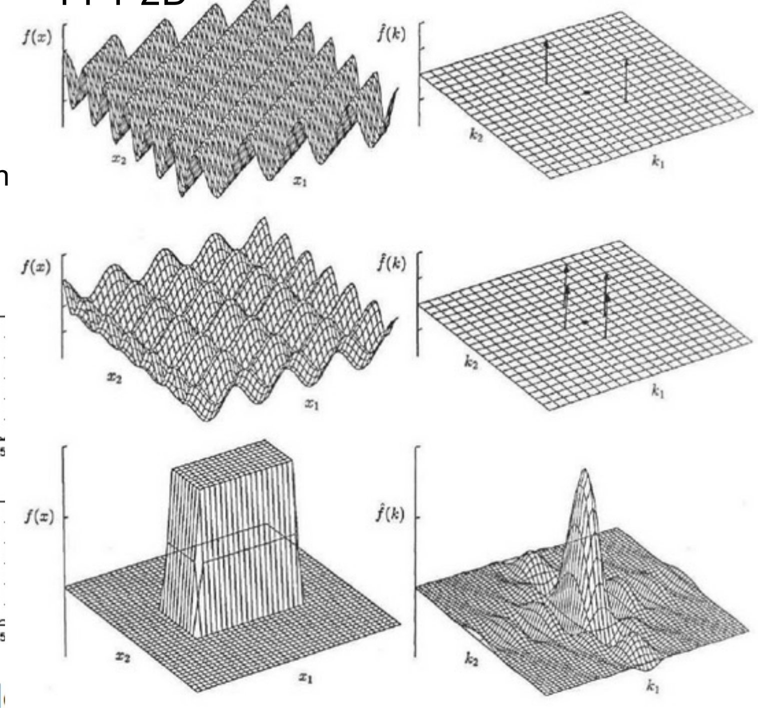


$|F(u,v)|$



Row and Column Profil

FFT 2D



Dataset

The dataset that we're using is a subset of BRATS
Brain Tumor Image Segmentation (BRATS) Data Set
BRATS is an international competition which focuses on advanced brain tumor detection and analysis
It provides updated MRI images with and without brain tumor
BRATS Challenge : https://www.med.upenn.edu/cbica/brats/

Steps

- Load all images using file handling from data set
- Change image into grayscale
- Apply FT and go into frequency domain
- Noise filtering
- Apply Inverse FT to go back into spacial domain
- Apply ML
- Discuss results

file handling

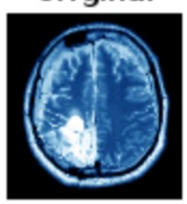
- Yes_images has all of the images with brain tumor and no_images without brain tumor
- Check length to verify all images are imported successfully
- Imread reads individual image files

```
clc;
yes_images=dir('yes/');
no_images=dir('no/');

x=length(yes_images);
y=length(no_images);

disp(x);
disp(y);

mri_image = imread('yes/Y258.JPG'); %read the MRI image
```

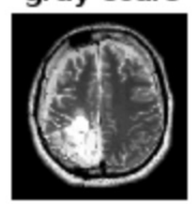
Change image into grayscale

- First change any colored image (Rgb) into gray scale
- notice the lower gray levels correspond to a greater amplitude (the brighter area is indicating a tumor)
- Rgb is an image with 3 color layers (red, green and blue respectively)
- Rgb2gray converts the 3-layered image into a single layered gray scale 2D image so that fourier transform can be easily applied
- Fourier transform cannot be directly applied to a 3-dimensional object (rows, columns and layers)

original



gray scale

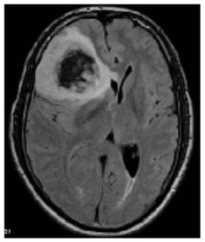


```
mri_image2= rgb2gray(mri_image);
```

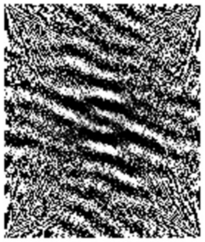
Apply FT

- Fourier transform is applied to MRI images to analyze the frequency component of the image data
- The fft2 function is used to plot the frequencies of the original image
- This function is an optimized version of the discrete fourier transform
- In the frequency domain, we can remove unwanted frequencies (noise) to improve the accuracy of machine learning algorithms

original



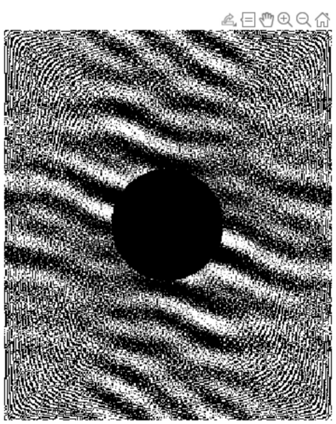
fourier image



```
fourier_image = fft2(mri_image2); %run fourier 2D
```

Noise filtering

- Nested for loop to apply low pass and high pass filter
- The center of the fourier image has frequency zero
- As we move away from the center, the amplitude of the frequency increases
- We are only removing frequencies that do not contribute to our image (noise)



```
cutoff_frequency_HP=50;
[m,n]=size(mri_image2);
filter_mask=ones(m,n);
for  i= 1:m
    for j=1:n
        distance=sqrt((i-m/2)^2+(j-n/2)^2); %distance from center
        if distance<=cutoff_frequency_HP
            filter_mask(i,j)=0; %high pass filter
        end
    end
end
```

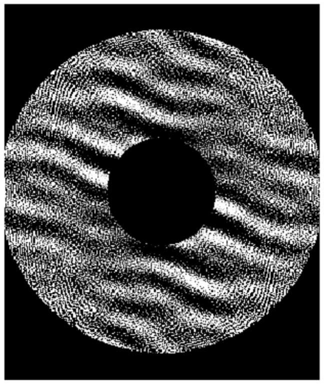
Noise filtering

- we apply both high pass and low pass to remove most of the unwanted frequencies

How do we indentify the optimal frequencies for high pass and low pass filtering?

magnitude spectrum

Peaks in magnitude spectrum indicate significant frequency components present in the fourier image



Noise filtering - for loops

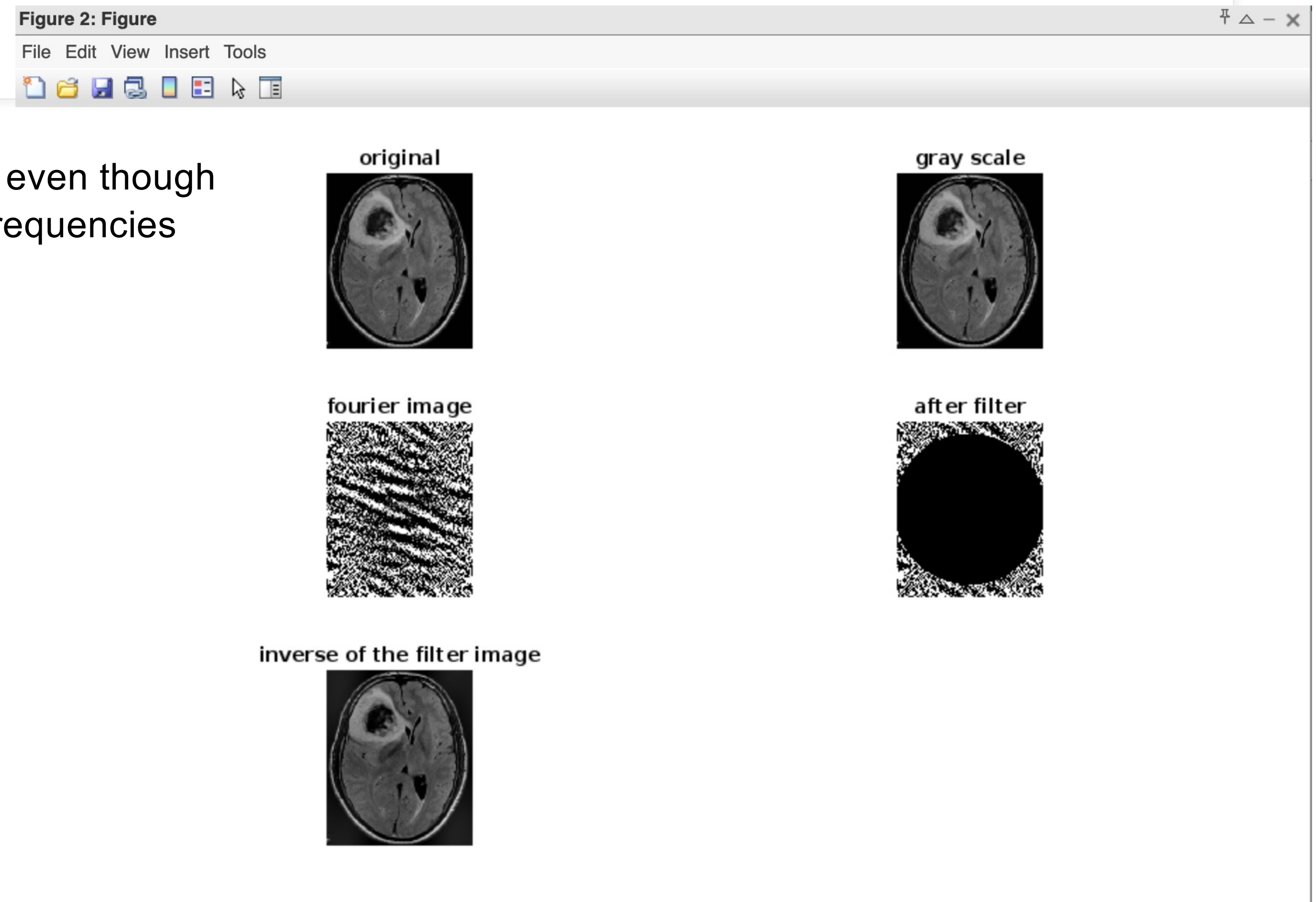
- I used for loop in making HP and LP filters
- Nested for loops are 2 for loops that will help visit each value in the 2D fourier image

Think of a fourier image as a Row x Column matrix and when you’re visiting each value you can filter them with the cutoff frequency

```
cutoff_frequency_LP=226.15;
[m,n]=size(mri_image2);
filter_mask_LP=zeros(m,n);
for  i= 1:m
    for j=1:n
        distance=sqrt((i-m/2)^2+(j-n/2)^2); %distance from center
        if distance<=cutoff_frequency_LP
            filter_mask_LP(i,j)=1; %low pass filter
        end
    end
end
```

Inverse fourier- IFFT2

- I still get the whole image back even though I have removed almost all the frequencies



Apply ML

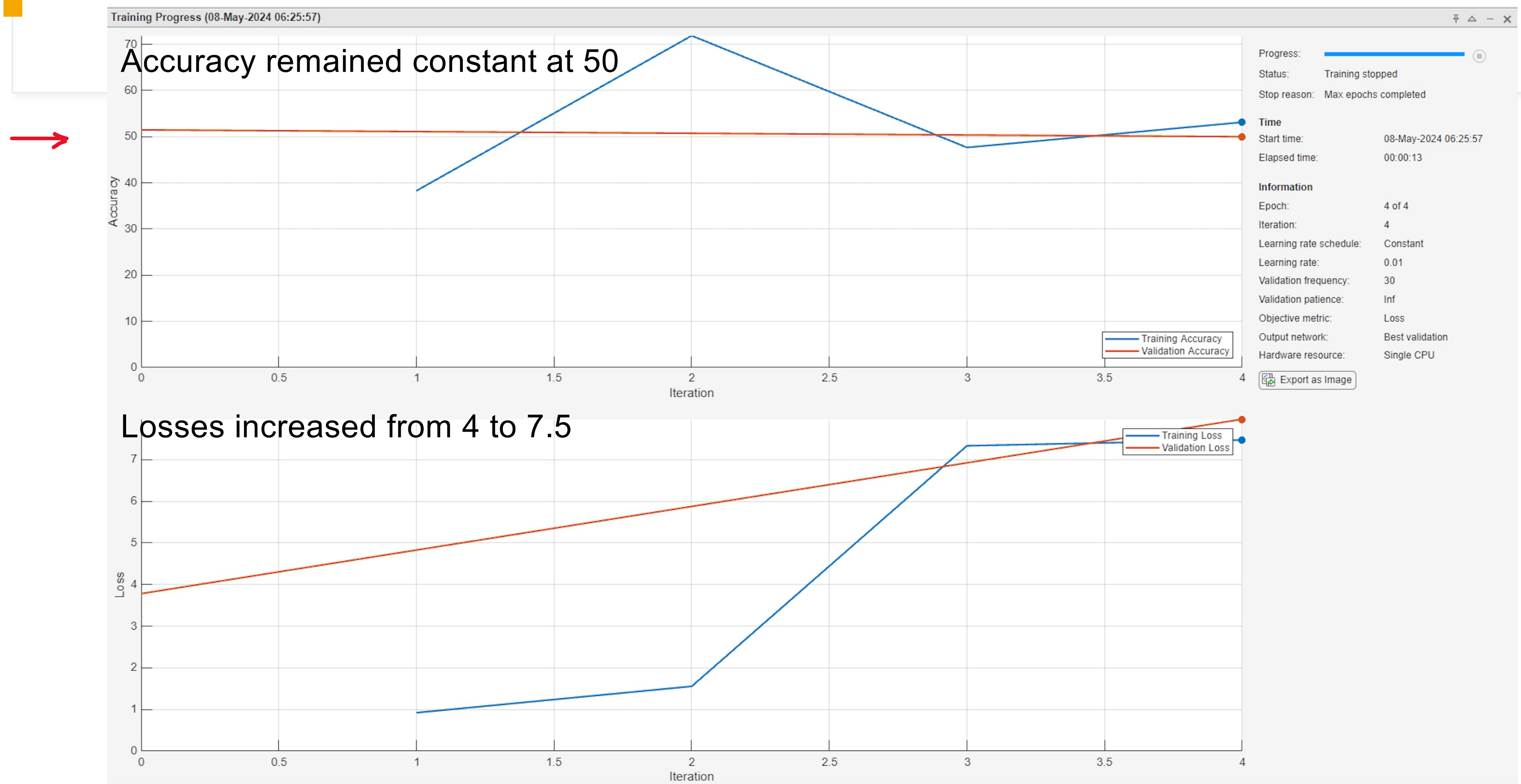
NNET is a free machine learning model which will classify new data into different groups then calculate its own accuracy

The code is provided with Yes and No folders, images with tumor and without tumor. So it knows the answer, first will train with 120 random images, this is called supervised learning, then it will apply its understanding to test the rest of the images

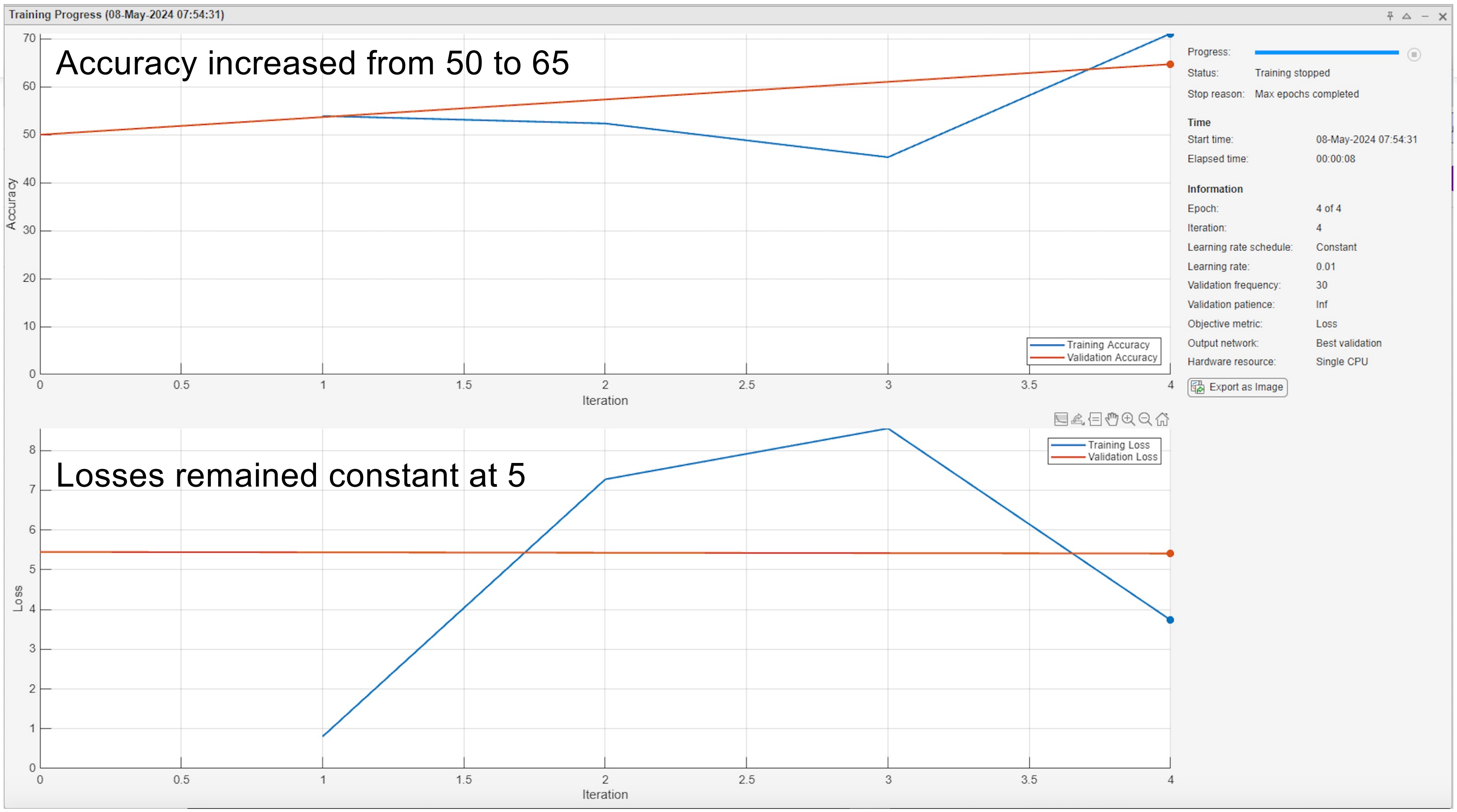
I will run the images that were subjected to FT then the original images that were not subjected to FT

we will run the code twice and the goal is to determine how many images have tumor and how many don't have tumor then calculate accuracy and compare results

ML accuracy for original MRI images (without fourier transform)



ML accuracy with fourier transform



Discuss results

Using fast fourier transform, I was able to increase the. accuracy of an ML model by 15% which resulted in improving the diagnostic value of MRI and to optimize automation