

## CS 396 Selected Topics in CS-2

### Research Project

Report Submitted for Fulfillment of the Requirements and ILO's  
for Selected Topics in CS-2 course for Fall 2021

Team ID No. 12

	ID	Name	Grade
1.	201900657	محمد حسن محمد القباني	
2.	201900659	محمد حسين محمد عبداللاه	
3.	201900066	احمد علاء الدين عبد الراضي	
4.	201900704	محمد عربي فتحي محمد	
5.	201900218	باهر تامر هاشم مصطفى	
6.	201900802	مريم عبد المجيد عبد السلام	

Delivered to:

**Dr. Wessam El-Behaidy**

**Eng. Salma Doma**

**Eng. Ahmed Nady**

## • Paper Details

- Paper Name: Multiclass classification of brain tumours using a novel CNN architecture.
- Authors : Hareem Kibriya & Momina Masood & Marriam Nawaz & Tahira Nazir .

Received: 3 September 2021 / Revised: 22 December 2021 /  
Accepted: 27 March 2022 # The Author(s), under exclusive licence  
to Springer Science+Business Media, LLC, part of Springer Nature  
2022

## • Project Description

Brain tumors are a deadly condition that radiologists have a tough time diagnosing. It is critical to make treatment-related decisions based on accurate and timely categorization of malignant cancers. Several approaches for detecting brain tumors have been presented in recent years. These strategies, however, necessitate handmade feature extraction and manual tumor segmentation prior to classification, which is error-prone and time-consuming.

**Datasets for it :** [Brain Tumor MRI Dataset](#)

It includes 8222 files for (Training & Testing)

### **Training (4 directories)**

- Glioma
- Meningioma
- pituitary
- no tumor

### **Testing (4 directories)**

- Glioma
- Meningioma
- pituitary
- no tumor

dimension of images = 512\*512px

## • Implementation details

According to the paper ,

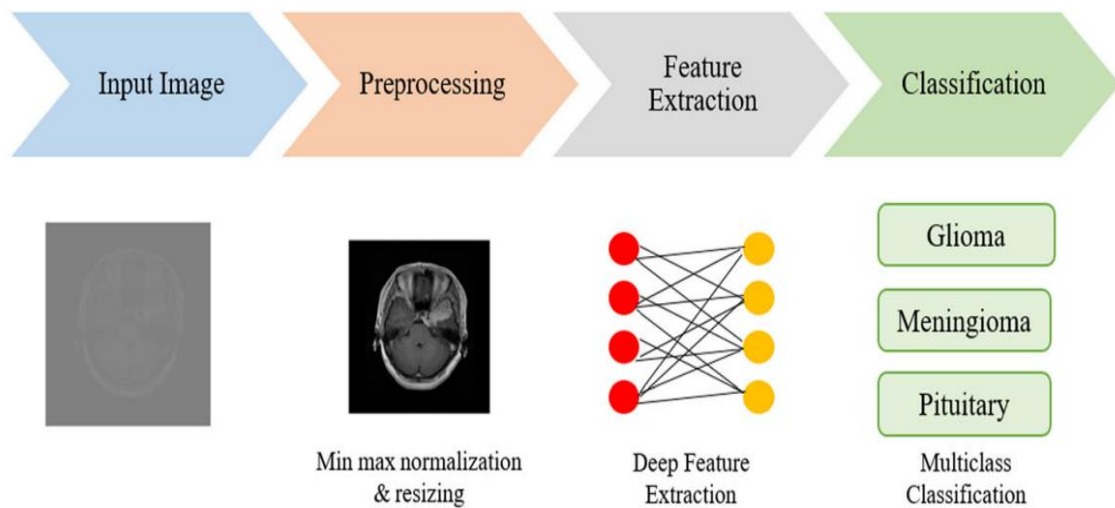
First we removed “No tumor” file from training and testing

Now number of files is : 7787

Then we split training data to (training and validation) with 30%

Now we have :

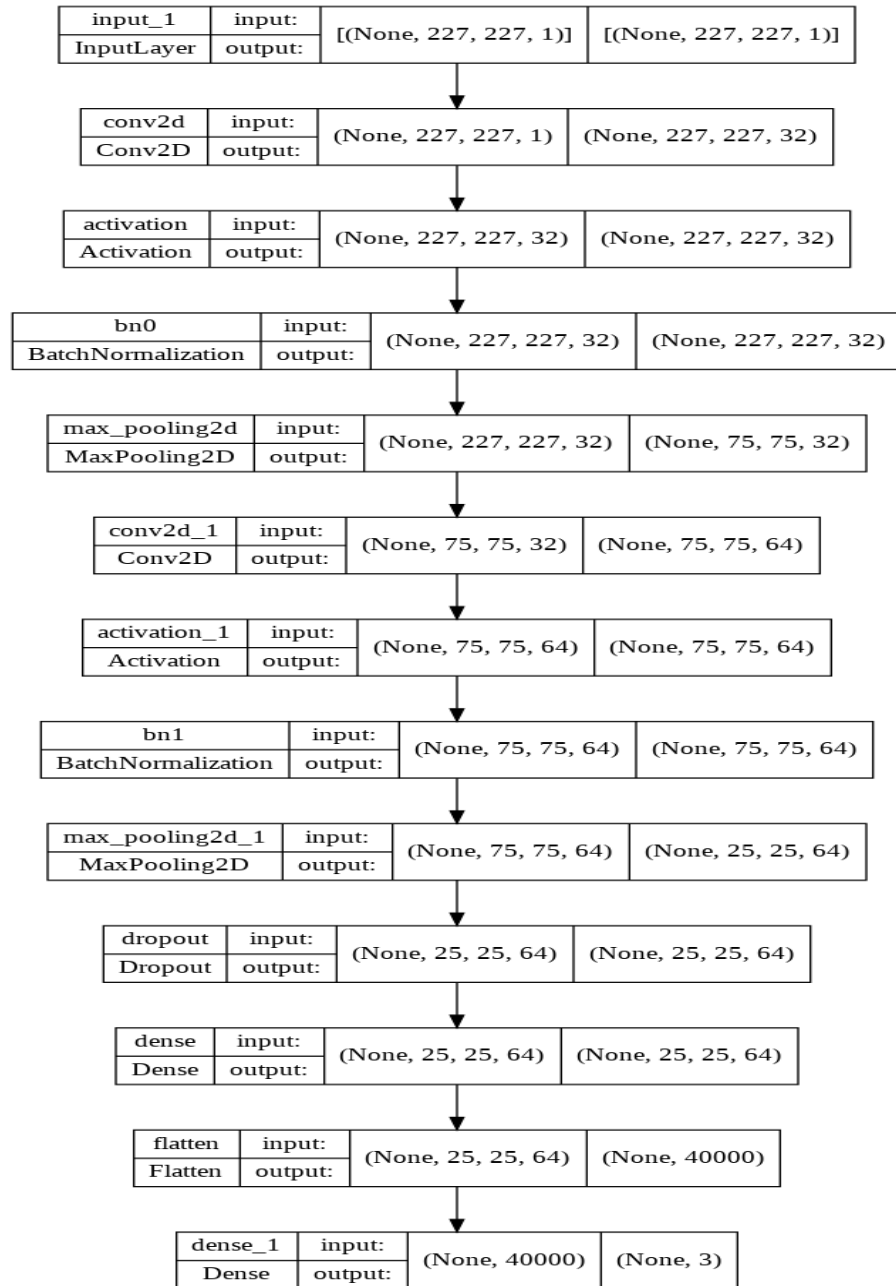
- Training (4614)
- Validation(1978)
- Testing (1195)



# • Our Model

13 layers

(input,2Conv,2Relu,2BatchNormalization,2MaxPool,Dropout,Fc,Softmax,Classification)

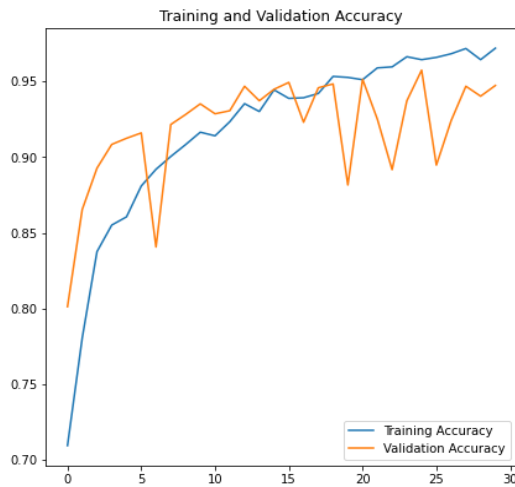


## Hyperparameters and architectures tested before reaching the final model

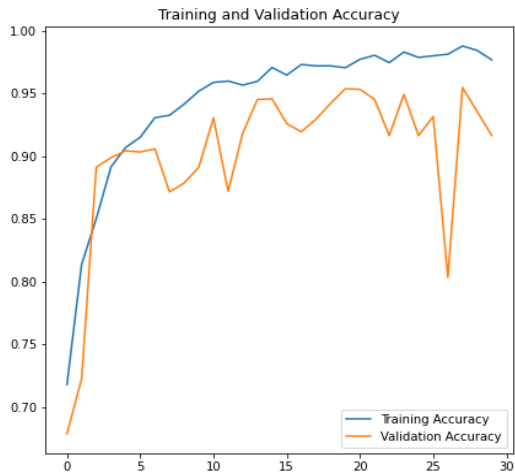
Factor (s)	Value (s)
Convolutional Layer+ReLU	1, <b>2</b> , 3
Batch Normalization Layer	1, <b>2</b> , 3
Dropout Layer	<b>1</b> , 2
Cross channel normalization layer	<b>0</b> , 1, 2
Fully Connected Layer	<b>1</b> , 2
Grouped Convolution Layers	<b>0</b> ,1, 2
Pooling Layers	<b>Max</b> , Average
Dropout Rate	0.25, <b>0.5</b>
Learning Rates	0.0001, 0.0002, <b>0.0003</b> , 0.0005
Optimizers	<b>Adam</b> , SGDM
No. of Epochs	10, 20, <b>30</b> , 40
Mini batch size	4, 8, <b>10</b>
No. of convolution kernels	<b>32</b> , <b>64</b> , 128, 256
Convolution kernel size	2, 3, <b>5</b>
Image Size	128×128, <b>227×227</b>

## • Testing results

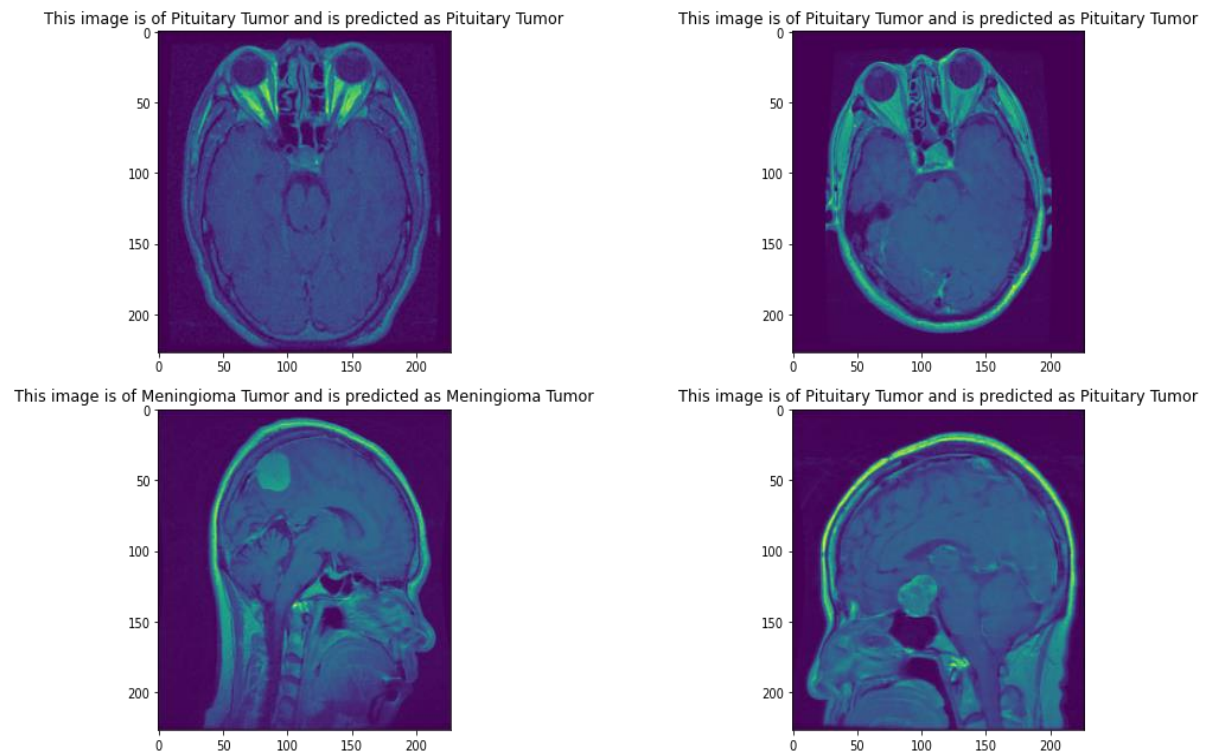
**When we keep Lr default and use Adam Optimizer:**



**When we change the learning rate to 0.0003 with adam Optimizer:**



## Outputs :



**Accuracy: 96 %**

```
▶ from keras.models import load_model
loaded_model = load_model("network.h5")
loss, accuracy = loaded_model.evaluate(x_test, y_test)

print('test accuracy : %.2f' % accuracy, "%")
```

```
↳ 62/62 [=====] - 1s 19ms/step - loss: 0.5322 - accuracy: 0.9585
test accuracy : 0.96 %
```



## Recall, Percision, f1-Score :

```
✓ [16] pred = model.predict(test_x)
45      pred = np.argmax(pred,axis=1)
      print(classification_report(test_y,pred))
```

	precision	recall	f1-score	support
0	0.93	0.77	0.84	400
1	0.79	0.94	0.86	421
2	0.96	0.93	0.94	374
accuracy			0.88	1195
macro avg	0.89	0.88	0.88	1195
weighted avg	0.89	0.88	0.88	1195

## Confusion matrix:

