

brain_tumor_investigation

October 15, 2022

0.1 CAPSTONE Project

0.2 Udacity AWS_ML nano degree

Hany Aly Elmogey

0.3 Project Overview:

Early detection and classification of brain tumors is an important research domain in the field of medical imaging and accordingly helps in selecting the most convenient treatment method to save patients life. Due to the increasing amounts of electronic data in the healthcare, medical doctors and physicians are facing problems in analyzing the data using traditional diagnosing systems. But with the help of machine learning a model can be trained to aid doctors and experts in detecting deadly diseases in their early stages. In this a Convolutional Neural Network machine learning model will be built and trained on Brain MRI Images Dataset.

0.4 Problem Statement:

The first step in machine learning problem is the dataset, how to collect the dataset, is it enough to build and train a machine learning model with acceptable accuracy, what is the type of the collected dataset and is it normally distributed and covering all the possibilities. In our case luckily Kaggle provides a good MRI images for 3 types of brain tumor ['glioma', 'meningioma', 'pituitary'] and no tumor so finally 4 labels. Will get into more details about the dataset later in this proposal.

The second step is preprocessing the data to prepare it to be feed to the machine learning model and get some insights for the data which will help during the training phase.

Third step is to build a suitable machine learning model using the suitable library, in our problem its a classification computer vision problem so I will use CNN also will try first to use transfer learning and use ResNet50.

0.5 Evaluation Metrics

I will use accuracy as our problem is a computer vision classification problem so accuracy will be a good representation of the model performance also will use the confusion matrix

as in my opinion its accepted to false classify a non tumor MRI as a tumor but not the other way around and the confusion matrix will be the best way to test the model performance in this case.

0.6 Data Exploration

0.6.1 To download the dataset click [here](#)

0.6.2 total number of training images is 5712 and divided on each type as follows:

number of images of tumor type “glioma” is 1321 number of images of tumor type “pituitary” is 1457 number of images of tumor type “notumor” is 1595 number of images of tumor type “meningioma” is 1339

0.6.3 total number of testing images is 1311 and divided on each type as follows:

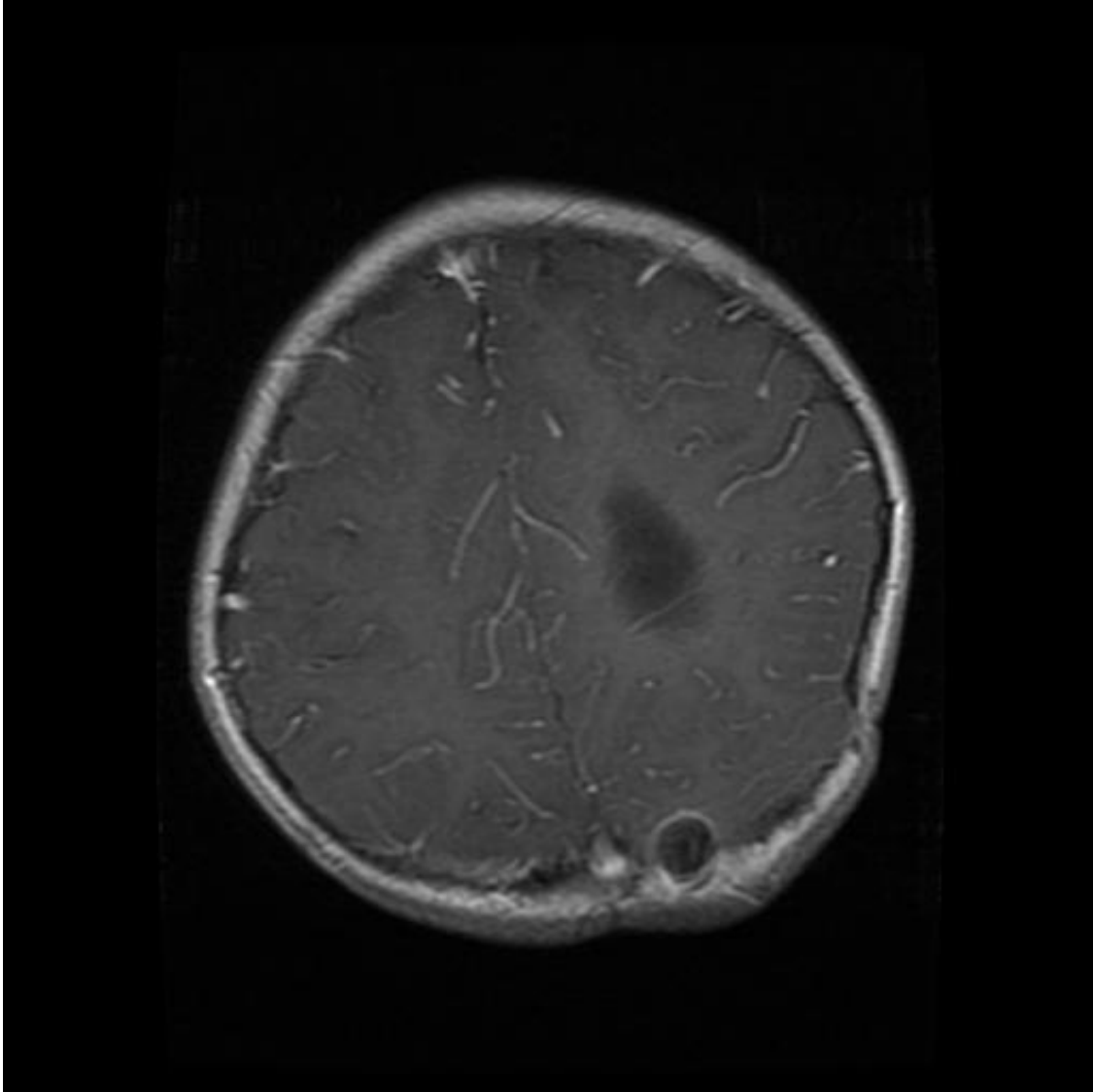
number of images of tumor type “glioma” is 300 number of images of tumor type “pituitary” is 300 number of images of tumor type “notumor” is 405 number of images of tumor type “meningioma” is 306

the images shape is 512*512 black and white.

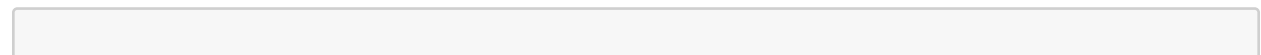
0.7 Data Visualization

[46]:

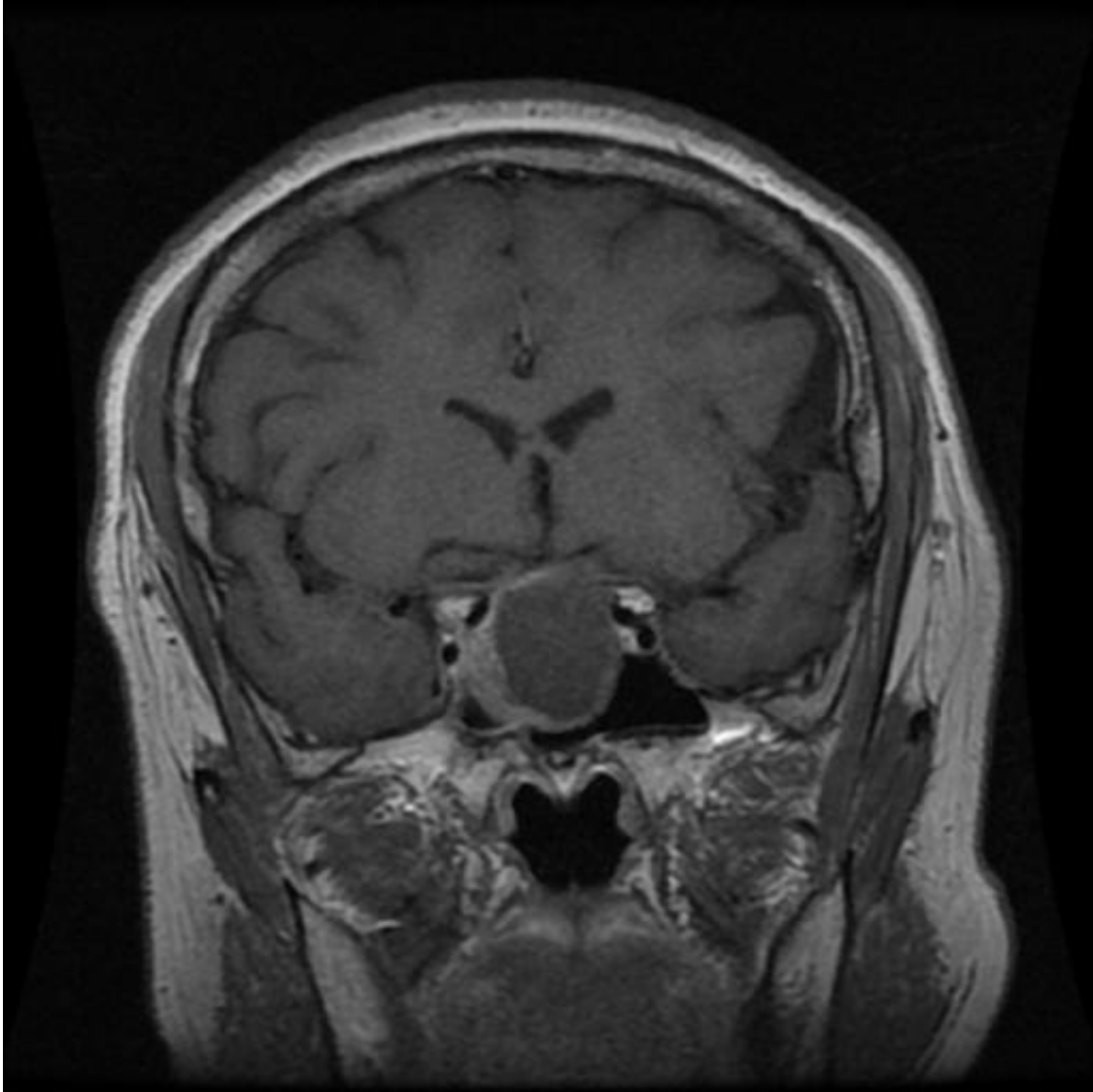
```
sample image for tumor type "glioma"  
(512, 512)
```



[48]:



sample image for tumor type "pituitary"
(512, 512)



[50]:

sample image for tumor type "meningioma"
(512, 512)



0.8 Benchmark Model:

brain tumor classification is performed by using Fuzzy C Means (FCM) based segmentation has been introduced in the following research paper which will be used as a benchmark to achieve the results <https://biomedpharmajournal.org/vol11no3/brain-tumor-classification-using-convolutional-neural-networks/>

0.9 Data Preprocessing

I will create a data loader that will load the images from the folders with their corresponding label using pytorch and resize the images to be 224*224 as its the input layer of the ResNet50 model and finally convert it to tensor.

0.10 Implementation

A convolutional neural network will be used and will use a ResNet50 pretrained model using pytorch. Sage maker notebook instance will be used with “ml.P2.xlarge” as instance type. Hyperparameters tuning will be used to try to reach the best model

0.11 Reflection

The problem was building a suitable model that classifies the brain Tumor, A data is used from Kaggle, The data set was good but may be more data could do some enhancement on the results. A CNN model using transfer learning is used, I used ResNet50 I guess other types of pretrained model as DenseNet-121 can be used to enhance the accuracy. AWS Sagemaker notebook instance will be used of the type “ml.P2.Xlarge” and hyperparameters tuning will be used to get the best model.

[]:

[]: