Intracranial Brain Hemorrhage Detection Small Research Project Spring 2020

1st Meghna Shrivastava
1st meghna3
Information Sciences

2nd Tafseer Khan
2nd tafseer2
Information Sciences

3rd Mohit Jain
3rd mohitj2
Electrical and Computer Engineering

***Abstract*—Intracranial Brain Hemorrhage is a type of health problem which causes bleeding in the skull. It is a severe health problem which has a high mortality rate of thirty-five percent. More than half of the deaths occurred within 24 hours from the time of diagnosis. Therefore, it is crucial to detect the presence of Hemorrhage with high accuracy.**

I. Introduction

In today's generation, medical imaging is absolutely necessary when tracking the progress of an ongoing illness. MRI's and CT scans allow the physician to monitor the effectiveness of treatment and adjust protocols as necessary. For our project too, we are dealing with a set of CT scanned images of the brain. Such CT scanned images helps the medical practitioners to identify whether a person is suffering from Hemorrhage or not. The main objective behind working on this project is to understand how to track and identify the main causes of the Hemorrhage within a brain and focus on improving the actual rate of identifying the images having Hemorrhages.

The current lifestyle of the people has led to a lot of internal ailments. Most of the time, it goes unnoticed. Sometimes, the incorrect diagnosis also lead to the wrong treatment that eventually proves fatal. Especially in Hemorrhage diagnosis, this has been widely seen. This is one of the main reasons that we chose to work on this project. We feel that medical practitioners if given a have better diagnosis platform might help them treat the patients much efficiently.

There has been a lot of research on this topic lately. The scope of this project can be extended to the multi-class classification of different Hemorrhage types, which gives a detailed insight to the type of ailment a person might suffer with. Such diagnosis also benefits the pharmaceutical companies that helps manufacture the drug to cure the ailment. This has been going since very long especially after the advancement in the technology.

What are we trying to achieve through this project? Our team focuses on identifying the main components that might help us achieve the following: 1. Understand the process of identifying the core components from medical images. 2. Developing a model using the identified core components that will help us classify the images having Hemorrhages with much better accuracy.

II. Data Insights

The dataset used here is given to us courtesy of Radiological Society of North America (RSNA) via Kaggle. The images were in DICOM format so we made use of the predefined Python library Pydicom and windowing method to extract the images onto a folder which can be used for further processing which were then converted to RGB images. This data processing is explained in the next section. The dataset has predefined labels for the classes as 0 for the Non-Hemorrhage cases and 1 for the Hemorrhage ones. If there is presence of hemorrhage, The data is again classified into five categories of Hemorrhage namely “intraparenchymal”, “intraventricular”,“subarachnoid”, “subdural”, and “epidural”.

In an attempt to simplify the project we made use of machine learning methods to predict only the presence of Hemorrhage or not without making prediction on the type of image.

The dataset[1] has a total of 674,023 images out of which 577,155 images are of the type ”Non-Hemorrhage” and the other 97,103 images are of the type Hemorrhage with further classification the five type of Hemorrhage mentioned above. The ratio of Non-Hemorrhage image to Hemorrhage image is approximately 6:1 which means that the dataset id very biased.

III. Data Prepossessing

A. Subset for balancing the dataset

From the figure 1 we can grasp the fact that the data is very skewed with the Non-Hemorrhage image taking a major portion of about 85% while Hemorrhage type occupy only 15%.
So we decided to achieve a 50-50 dataset in order to have meaningful insights from the results of the machine learning processes. In an attempt of dong so we selected random set of about 97,103 images from the Non-Hemorrhage set. After the segregation was done we had a total subset of about 194,206 sample images.

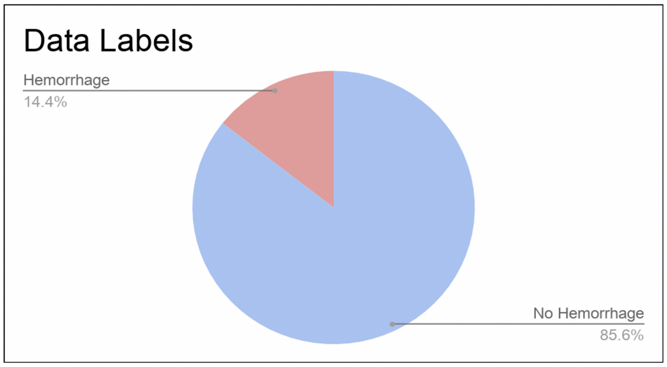

Fig. 1. Ratio of Dataset Labels

B. Subset for Faster Modelling

Processing a set of about 200K images was a difficult task. In an attempt to make our processing faster we randomly sampled out about 15K images from the dataset. The prepossessing and the machine learning methods were applied to these images only.

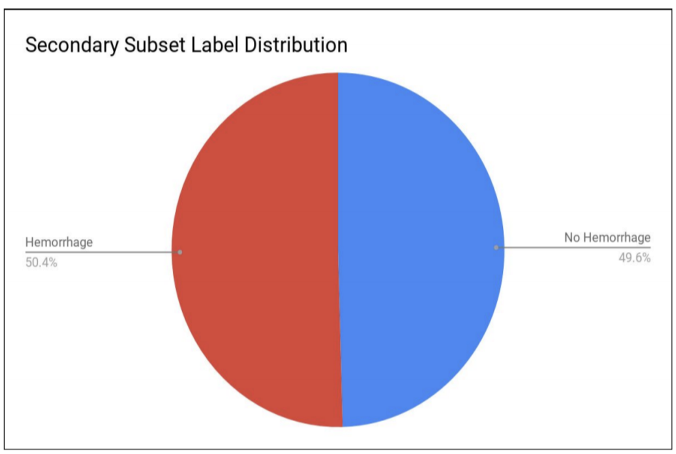

Fig. 2. Ratio of Subset

C. Image Processing

Since we are dealing with the CT scanned images in this project, we had to come up with a process that would help convert the CT images into the RGB image. Process involves identifying the unit measure of the CT images and understanding how the core components within the CT scanned images range in terms of Hounsfield unit.

The original dataset provided by RSNA contains only DICOM files. We cannot directly feed the DICOM file into the neural network models. Thus, it is necessary to transform those files into RGB .JPG format which can be extracted into array forms for models. Unlike regular images which use RGB to distinguish different items, DICOM format uses Hounsfield Unit (HU) which is a measurement of radio density. Different matters in brains have different HU which can be used to represent the existence of intracranial hemorrhage. According to NCBI, HU for human brain ranges from -1,000 to +1,000. However, traditionally, CT scan images use gray scales ranging from zero to 255, which cannot represent all HU values. Therefore, we should indicate the range of HU in which we are interested, so that we can use gray scales to represent the interested range. It is the process of choosing windows, or windowing. The windowing mechanism is shown in

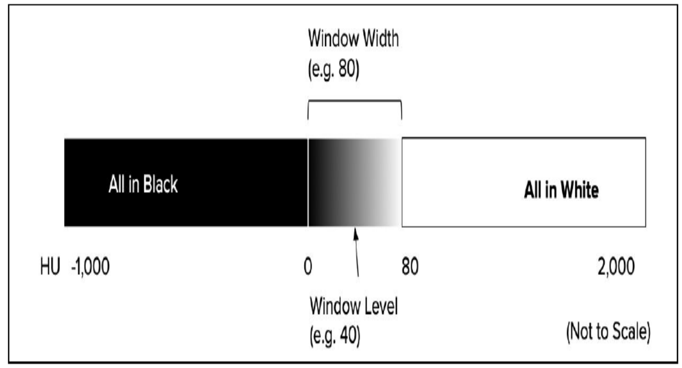

Fig. 3. Windowing

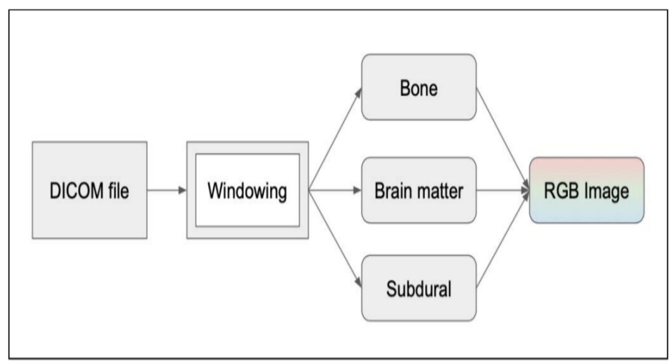

Fig. 4. Image Processing

If we only need to focus on one matter such as bone, we can set up the window center to the HU of bones. The output gray scale images will only show the rich features of bone, while matters with HU far from the HU of bones will either appear white or black. As we referred to an article on the radiology forum, there are three important windows for detecting intracranial hemorrhage, which are bone, brain matter, and subdural windows. We extracted each window from the DICOM files to one color channel. Then, we formed the three color channels into JPG images. Because there are three channels and each channel represents one important window, the output JPG images are RGB images. By forming RGB images, one of the advantages is that the neural network modeling can be simplified. We can directly feed the RGB images into the model without losing any essential features from the DICOM files. The process is visualized in Figure 4.

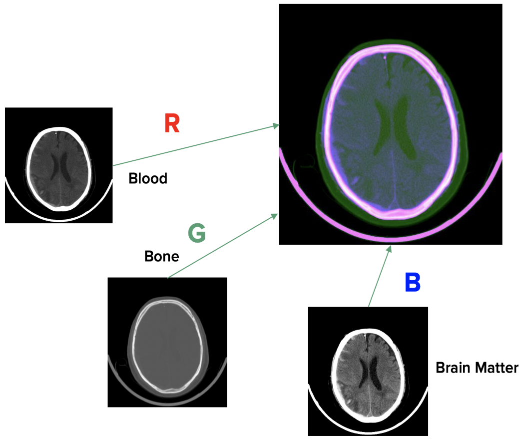

Fig. 5. RGB image

In Figure 5, we can see 3 separate gray scale images showing main components i.e. Blood, Bone and Brain Matter. These 3 gray scaled images are obtained after performing windowing technique on a CT scan image. But then the question arises, how do we merge these three gray scale images into one RGB image? After researching a bit, we came up with a solution as shown in Figure 5. We assigned one channel to each gray scale image. There are 3 active gray scale images and 3 channels i.e Red, Blue and Green. So, we assigned each channel to one gray scale image as shown in the Figure. This led to the generation of an RGB image as seen above.

IV. Feature Extraction

The image in the dataset have a dimension of 512x512 which counts to about 262,144 pixels per image. So in order to reduce the computation time we perform 3 methods of feature extractions namely-

A- Local Binary Pattern
B- Gray Level Co-Occurrence Matrix
C- Gabor Filter

The Flow for feature extraction is given below-

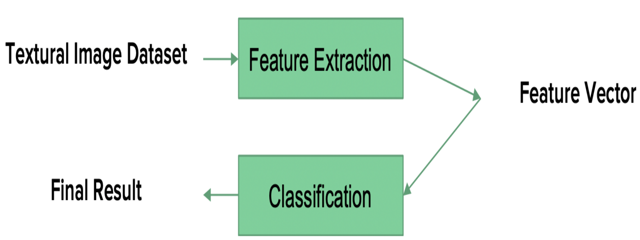

Fig. 6. Flow for feature extraction

A. Local Binary Pattern

Local Binary Pattern is a very simple image feature extraction method which derives the histogram of the image pixel values and this histogram is used for further classification.

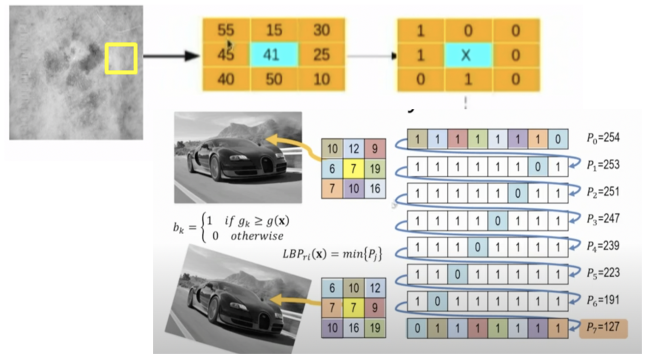

Fig. 7. Linear Binary Pattern

In Local Binary Pattern a matrix from the image is chosen let it be a 3x3 matrix. The central element is chosen as the reference. All the surrounding points which have a value more than or equal to the central element are assigned a value of 1 and the others are assigned 0.
These values are then kept in a one dimension array and the minimum decimal value is then calculated to form the histogram. This histogram is the feature vector which is used by Support Vector Machine for the purpose of classification.

B. Gray Level Co-Occurrence Matrix

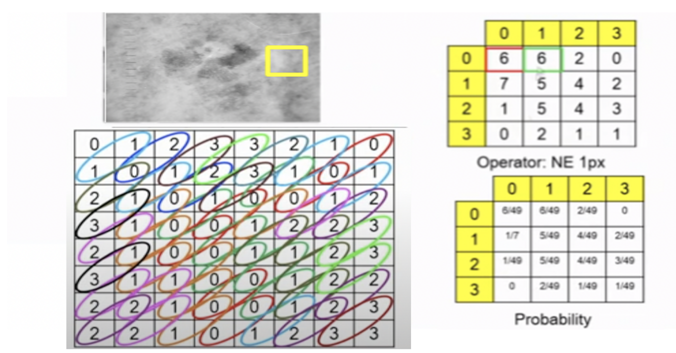

Fig. 8. Gray Level Co-Occurrence Matrix

GLCM is a statistical method that makes use of spatial relationship of pixels to extract the features of the image. In this method part of the image matrix is taken into consideration. Then a NxN matrix is created where N is max value in the image matrix. Every element in the matrix is compared to the right diagonal element and the count is updated in the matrix that we have created. Then the values are updated based on the probability of each combination happening. This matrix is then used for feature extraction by SVM.

C. Gabor Filter

Gabor filters are band pass filters that are used to extract a specific band from the images.

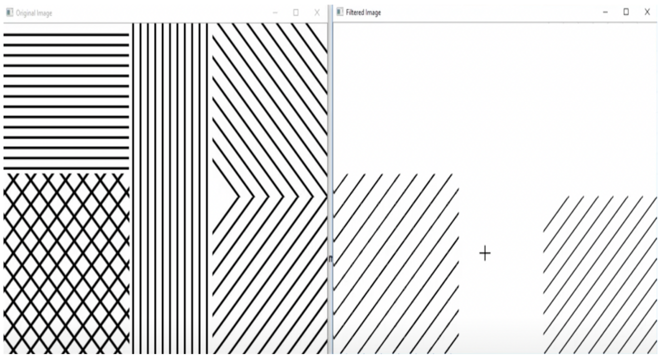

Fig. 9. Gabor Filter as Band Pass Filter

Gabor filter response is used to find the presence of any frequency component in a specific direction in the selected window of the image. In the above image the Gabor filter is configured so as to obtain only the specific slanted parallel lines and none of the other part of the image.
We used A bank of 16 Gabor filter oriented at an angle of 11.250 (i.e. if the first filter is at 00, then the second will be at 11.250, the third will be at 22.50, and so on.)

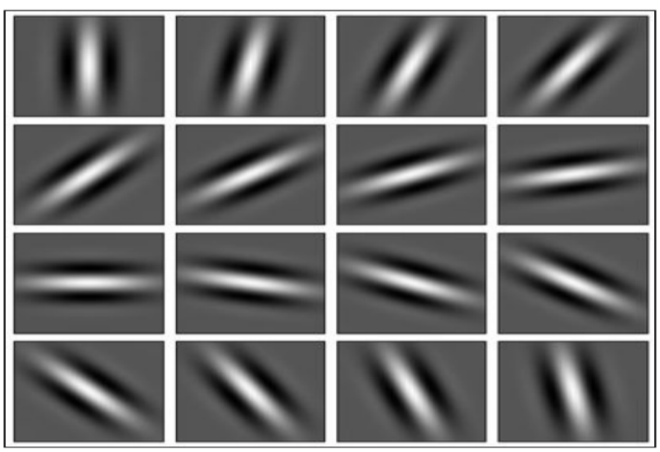

Fig. 10. A bank of 16 Gabor Filter oriented at 11.25 degree from one another

V. Machine Learning Models and Results

A. Support Vector Machine

The next step after Feature Extraction is to pass the Extracted feature for binary Classification. We decided to make use of SVM for our classification purposes. The method was implemented with 5 fold Cross Validation. We found C=100 to be an optimum parameter for our model. With a train test split of 80:20 we archived an accuracy of 72.06% with GLCM.

The results from comparison of the various Feature Extraction methods are shown in the table-

	Local Binary Pattern	Grey Level Co-occurrence Matrix (GLCM)	Gabor Filter
Accuracy	66.76%	72.06%	61.11%
Precision	67.60%	71.13%	60.97%
Recall	66.57%	76.34%	58.27%

Fig. 11. Results of Feature Extarction Methods

From the results shown in the table purpose of binary classification the best performance was obtained by Gray Level Co-Occurrence Matrix.

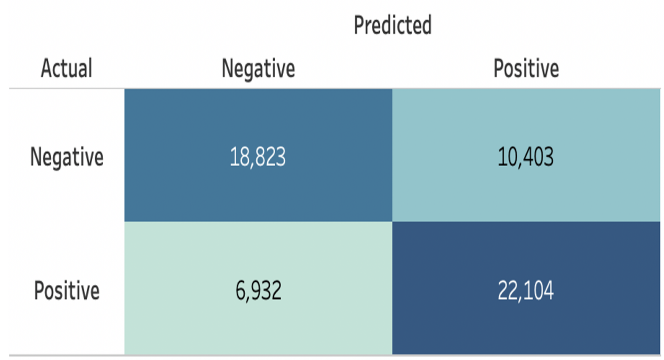

Fig. 12. Confusion Matrix for SVM

The above image shows the confusion matrix for SVM model. As we can see that we have high values for False Positive and False Negatives also. This is because there is equal amount of images with Hemorrhage as there is images with Non-Hemorrhage. If we would have used the ratio of the original dataset then we would have had comparatively less mispredictions.

B. CNN and its Results

We also used Keras to build a Convolutional Neural Network model. The secondary subset was used for faster modeling due to time limitation. We split the dataset into training set and validation set by the ratio of eight to two. In Table 1, the data size of each label and each set is listed.

At the early stage of the model building, we used pandas data frames to store the image dataset, where we faced the “out of memory” error. We then rewrote the script to adopt flow from directory method which generates batches to decrease memory usage, which avoided the memory error.

Table 1. Data Distribution in the CNN Model

Labels	Training Set	Validation Set	Ratio of Labels
Non-hemorrhage	5924	1427	49.6%
Hemorrhage	5939	1544	50.4%

Fig. 13. Data Distribution for CNN

During the model fitting stage, we used batch size as 32 and epoch as 20. The accuracy is illustrated in Figure 14, while the loss is shown in Figure 15.

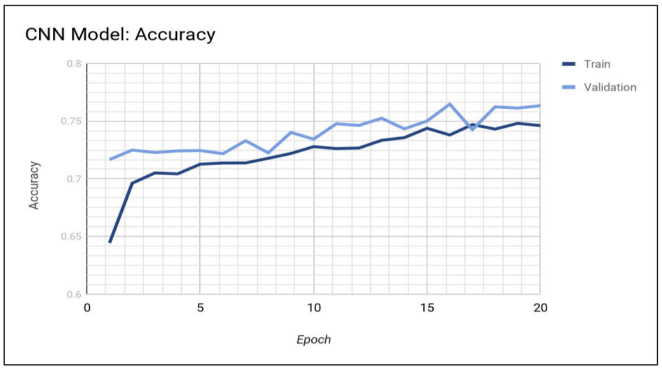

Fig. 14. Accuracy for CNN

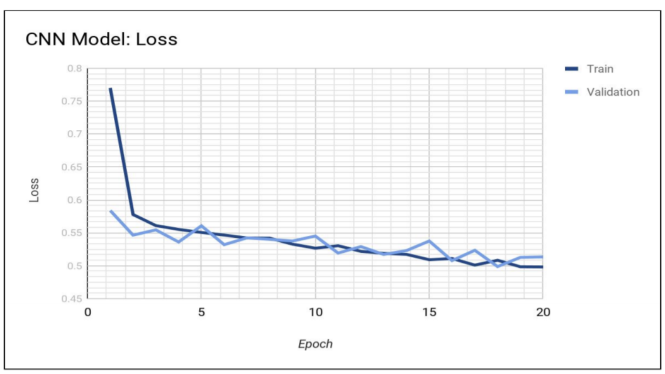

Fig. 15. Loss for CNN

VI. Conclusion and Future Scope/Improvements

Throughout the project, we divided the process it into several parts in terms of different models and data pre-processing. All the 3 teammates were equally involved in each phase of the project and focused heavily on building the model and pre-processed the datasets. Our focus was also on extracting the important features from the images. In addition, we worked on the SVM models and streamlined the CNN models together. By using the instance with a powerful GPU, we were able to expedite the process too. Moreover, being more familiar with Ubuntu and GPU setup are also essential. There are some further recommendations for future studies. 1. Object detection: The process can be improved by concentrating on the spots that have intracranial hemorrhage. It can potentially improve the accuracy if the hemorrhage areas on images can be pointed on. 2. Remove low quality CT images: Some images are poorly taken which can hurt the training process. The images which appear all in black or white without brain information can be manually removed. 3: Enlarge training set Due to time limitation, we took subset from the original dataset. By including more data we can improve the performance.

References

[1] "(https://www.kaggle.com/c/rsna-intracranial-hemorrhage-detection)."

[2] "The melanoma skin cancer detection and classification using support vector machine." *Hiam Alquran1, *, Isam Abu Qasmieh1, Ali Mohammad Alqudah1, Sajidah Alhammouri1, Esraa Alawneh1, Ammar Abughazaleh1, Firas Hasayen1.*

[1] and [2]