1. INTRODUCTION

1.1 Overview:

Brain tumor is one of the most rigorous diseases in medical science. Effective and efficient analysis is always a key concern for radiologists in the premature phase of tumor growth. Histological grading, based on a stereotactic biopsy test, is the gold standard and the convention for detecting the grade of a brain tumor. The biopsy procedure requires the neurosurgeon to drill a small hole into the skull from which the tissue is collected. There are many risk factors involving the biopsy test, including bleeding from the tumor and brain causing infection, seizures, severe migraine, stroke, coma, and even death. But the main concern with the stereotactic biopsy is that it is not 100% accurate which may result in a serious diagnostic error followed by wrong clinical management of the disease.

Tumor biopsy is challenging for brain tumor patients, non-invasive imaging techniques like Magnetic Resonance Imaging (MRI) have been extensively employed in diagnosing brain tumors. Therefore, the deployment of systems for the detection and prediction of the grade of tumors based on MRI data has become necessary. But at first sight of the imaging modality like in Magnetic Resonance Imaging (MRI), the pre visualizations of the tumor cells and its differentiation with its nearby soft tissues is a somewhat difficult task which may be due to the presence of low illumination in imaging modalities or its large presence of data or several complexity and variance of tumors-like unstructured shape, viable size, and unpredictable locations of the tumor.

1.2 BRAIN ANATOMY:

The brain tumor is one all the foremost common and, therefore, the deadliest brain disease that has affected and ruined several lives in the world. Cancer is a disease in the brain in which cancer cells ascend in brain tissues. Conferring to a new study on cancer, more than one lakh people are diagnosed with brain tumors every year around the globe. Regardless of stable efforts to overcome the complications of brain tumors, figures show unpleasing results for tumor patients.

To contest this, scholars are working on computer vision for a better understanding of the early stages of tumors and how to overcome using advanced treatment options.

Magnetic resonance (MR) imaging and computed tomography (CT) scans of the brain are the two most general tests to check the existence of a tumor and recognize its position for progressive treatment decisions. These two scans are still used extensively for their handiness, and the capability to yield high-definition images of pathological tissues is more. At present, there are several other conducts offered for tumors, which include surgery, therapies such as radiation therapy, and chemotherapy. The decision for which treatment relies on the many factors such as size, kind, and grade of the tumor present in the MR image. It's conjointly chargeable for whether or not cancer has reached the other portions of the body.

Precise sighting of the kind of brain abnormality is enormously needed for treatment operations with a resolution to diminish diagnostic errors. The precision is often makeshift utilizing computer-aided diagnosis (CAD) systems. The essential plan of computer vision is to produce a reliable output, which is an associate estimation to assist medical doctors in image understanding and to lessen image reading time. These advancements increase the steadiness and correctness of medical diagnosis — however, segmenting an MR image of the tumor and its area itself a very problematic job. The occurrence of tumors in specific positions within the brain image without distinguishing picture intensities is an additional issue that makes a computerized detection of brain tumor and segmentation a problematic job.

1.3 Brain Tumor

According to Ilhan et al. [2], a brain tumor occurs when abnormal cells form within the brain. Many different types of brain tumors exist. Some brain tumors are noncancerous (be-sign), whereas some brain tumors are cancerous (malignant) and some are pre-malignant. Cancerous tumors can be divided into primary tumors that start within the brain, and secondary tumors that have spread from somewhere else, known as brain metastasis tumors.

1.4 Classification of Brain Tumor

There are two types of brain tumors. One is a Benign Tumor characterized as non-cancerous and the other one is Malignant Tumor- also known as Cancerous Tumor. 1.3.1 Benign Tumor Benign brain tumors are usually defined as a group of similar cells that do not follow normal cell division and growth, thus developing into a mass of cells that microscopically do not have the characteristic appearance of cancer [3]. These are the properties of a benign tumor:

• Most benign tumors are found by CT or MRI brain scans.

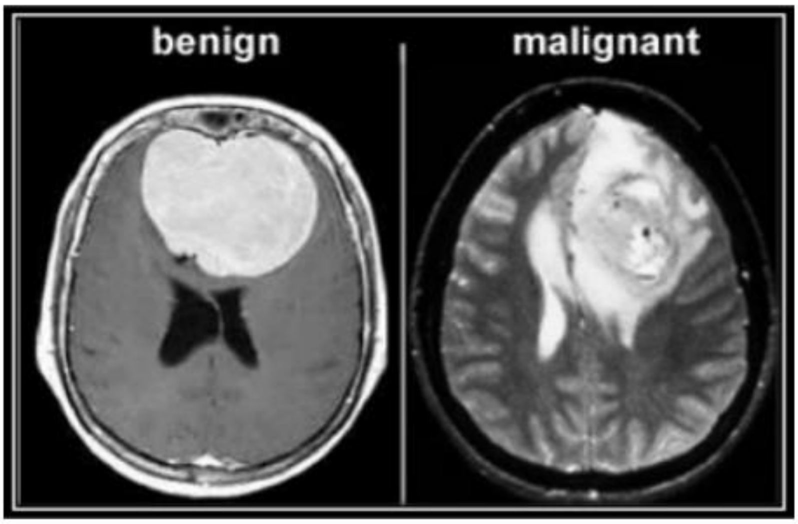
• Grows slowly, does not invade surrounding tissues or spread to other organs
has a border or edge that can be seen on CT scans.

• It can be life-threatening because it can compress brain tissues and other structures inside the skull, so the term 'benign' can be misleading. 1.3.2 Malignant Tumor Malignant brain tumors contain cancer cells and often do not have clear borders. They are considered to be life-threatening because they grow rapidly and invade surrounding brain tissues [4]. These are the properties of a malignant tumor:

• Fast-growing cancer that spreads to other areas of the brain and spine.

• A malignant brain tumor is either graded 3 or 4, whereas grade 1 or 2 tumors are usually classified as benign or non-cancerous.

• Generally these are more serious and often more fatal threats to life



.

Figure 1.1: Benign Tumor (left) and Malignant Tumor (Right)

1.5 MOTIVATION FOR THE WORK

A brain tumor is defined as an abnormal growth of cells within the brain or central spinal canal. Some tumors can be cancerous thus they need to be detected and cured in time. The exact cause of brain tumors is not clear and neither is the exact set of symptoms defined, thus, people may be suffering from it without realizing the danger. Primary brain tumors can be either malignant (contain cancer cells) or benign (do not contain cancer cells).

Brain tumors occurred when the cells were dividing and growing abnormally. It is appearing to be a solid mass when it is diagnosed with diagnostic medical imaging techniques. There are two types of brain tumor which is primary brain tumors and metastatic brain tumors. A primary brain tumor is a condition when the tumor is formed in the brain and tended to stay there while a metastatic brain tumor is a tumor that is formed elsewhere in the body and spread through the brain.

The symptom of a brain tumor depends on the location, size, and type of the tumor. It occurs when the tumor compresses the surrounding cells and gives out pressure. Besides, it is also occurring when the tumor blocks the fluid that flows throughout the brain. The common symptoms are headache, nausea, vomiting, and problems in bncing and walking. Brain tumors can be detected by diagnostic imaging modalities such as CT scans and MRIs. Both of the modalities have advantages in detecting depending on the location type and the purpose of examination needed. In this paper, we prefer to use MRI images because it is easy to examine and gives out accurate calcification and foreign mass location.

The MRI is the most regularly utilized strategy for imaging brain tumors and the identification of its vicinity. The conventional technique for CT and MR image classification and detection of tumor cells remains largely supported for human reviewing apart from different other methods. MR images are mainly used because there are non-destructive and non-ionizing. MR imaging offers high-definition pictures that are extensively utilized in discovering brain tumors. MRI has diverse schemes such as flair, T1-weighted, and T2-weighted images. There are many image processing techniques.

such as pre-processing, segmentation of images, image improvements, feature extraction, and classifiers.

Observing the recent statistics of the death rate caused by brain tumors, we selected brain tumor detection and classification which belongs to the field of medical image analysis. Tumor detection in medical images is time-consuming as it depends on human judgment. The experts in this field, such as radiologists, and specialized doctors examine CT scans, MRI, and PET scan images and give decisions upon which the treatment depends. This whole process is time-consuming. Automated medical image analysis can help to reduce the time and effort taken here and the workload of a human as it will be done by machines.

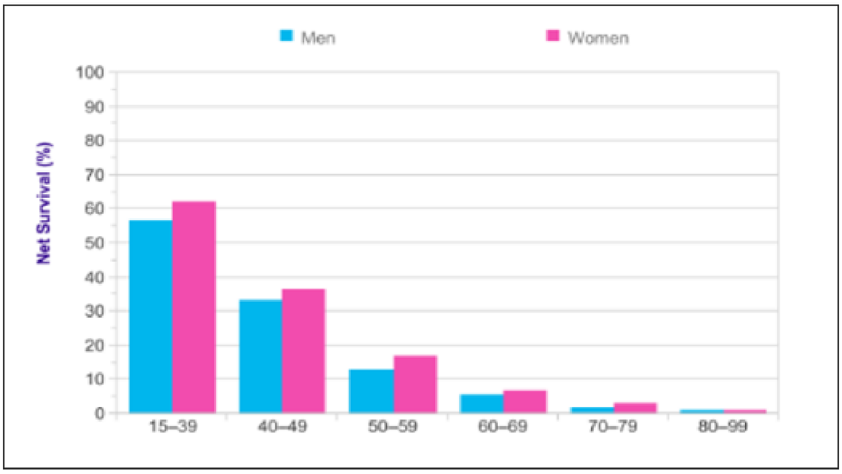


Figure 1.2: New cases and survival rate caused by a brain tumor

Figure 1.2 shows that death caused by brain cancer is higher than other types of cancers.

Brain tumor detection in an early stage can help to reduce the death rate in this field. For supporting faster communication, where patient care can be extended to remote areas using information technology, automated image analysis will help to a great extent.

The developed countries of the world have been introduced to the automation of medical image analysis. But, in Bangladesh, it has not been adopted well yet. We want to build a model which will be efficient and feasible from the perspective of Bangladesh. If we go through the most recent decade's statistics, it had been shown that there was an estimated 14.1 million cancer cases around the world in 2012. Men comprised 7.4 million among them, while the rest 6.7 million consisted of female females number is expected to increase to 24 million by 2035. Among all the forms of cancers, lung cancer was the most common cancer worldwide contributing 13% of the total .

1.5 Purpose

Automated defect detection in medical imaging using machine learning has become an emergent field in several medical diagnostic applications. Its application in the detection of brain tumors in MRI is very crucial as it provides information about abnormal tissues which is necessary for planning treatment. Studies in the recent literature have also reported that automatic computerized detection and diagnosis of the disease, based on medical image analysis, could be a good alternative as it would save radiologist time and also obtain a tested accuracy.

2. LITERATURE SURVEY

In Medical diagnosis, robustness and accuracy of the prediction algorithms are very important, because the result is crucial for treatment of patients. There are many popular classification and clustering algorithms used for prediction. The goal of clustering a medical image is to simplify the representation of an image into a meaningful image and make it easier to analyze. Several Clustering and Classification algorithms are aimed at enhancing the prediction accuracy of the diagnosis process in detecting abnormalities.

In the literature survey, we provide a brief summary of the different methods that have been proposed for clustering over the period of 2002 to 2018. We have been through 25 papers each of which has a unique approach toward segmentation in some parameter or the other. The summaries of each of the papers are provided below.

- **Siva Ramakrishnan And Dr. M. Karnan "A Novel Based Approach for extraction Of Brain Tumor In MRI Images Using Soft Computing Techniques," International Journal Of Advanced Research In Computer And Communication Engineering, Vol. 2, Issue 4, April 2013.**

A. Sivaramakrishnan et al. (2013) projected an efficient and innovative discovery of the brain tumor vicinity from an image that turned into a finish using the Fuzzy Approach grouping algorithm and histogram equalization. The disintegration of images is achieved by the usage of principal factor evaluation to reduce the extent of the wavelet coefficient. The outcomes of the anticipated FCM clustering algorithm accurately dream or area from the MR images.

- **Asra Aslam, Ekram Khan, M.M. Sufyan Beg, Improved Edge Detection Algorithm for Brain Tumor Segmentation, Procedia Computer Science, Volume 58,2015, Pp 430-437, ISSN 1877-0509.**

M. M. Sufyan et al. have presented a detection using an enhanced edge technique for brain tumor segmentation that mainly relied on Sobel feature detection. Their presented work associates the binary thresholding operation with the Sobel approach and excavates diverse extents using secure contour process. After the completion of that process, cancer cells are extracted from the obtained picture using intensity values.

- **B.Sathya and R.Manavalan, Image Segmentation by Clustering Methods: Performance Analysis, International Journal of Computer Applications (0975 – 8887) Volume 29– No.11, September 2011.**

Sathya et al. (2011) provided different clustering algorithm such as K-means, Improvised K-means, C-means, and improvised C-means algorithms. Their paper presented an experimental analysis for massive datasets consisting of unique photographs. They analyzed the discovered consequences using numerous parametric tests.

- **Devkota, B. & Alsadoon, Abeer & Prasad, P.W.C. & Singh, A.K. & Elchouemi, A. (2018). Image Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction. Procedia Computer Science. 125. 115-123. 10.1016/j.procs.2017.12.017.**

B. Devkota et al. have proposed that a computer-aided detection (CAD) approach is used to spot abnormal tissues via Morphological operations. Amongst all the different segmentation approaches existing, the morphological opening and closing operations are preferred since it takes less processing time with the utmost efficiency in withdrawing tumor areas with the least faults.

- **K. Sudharani, T. C. Sarma and K. Satya Rasad, "Intelligent Brain Tumor lesion classification and identification from MRI images using a K-NN technique," 2015 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kumaracoil, 2015, pp. 777-780. DOI: 10.1109/ICCICCT.2015.7475384**

K. Sudharani et al. presented a K- nearest neighbor algorithm to the MR images to identify and confine the hysterically full-fledged part within the abnormal tissues. The proposed work is a sluggish methodology but produces exquisite effects.

- **Kaur, Jaskirat & Agrawal, Sunil & Renu, Vig. (2012). A Comparative Analysis of Thresholding and Edge Detection Segmentation Techniques. International Journal of Computer Applications.vol. 39.pp. 29-34.10.5120/4898-7432.**

Jaskirat Kaur et al. (2012) defined a few clustering procedures for the segmentation process and executed an assessment on distinctive styles for those techniques. Kaur represented a scheme to measure selected clustering techniques based on their steadiness in exceptional tenders. They also defined the diverse performance metric tests, such as sensitivity, specificity, and accuracy.

- **Li, Shutao, JT-Y. Kwok, IW-H. Tsang and Yaonan Wang. "Fusing images with different focuses using support vector machines." IEEE Transactions on neural networks 15, no. 6 (2004): 1555-1561.**

J.T. Kwok et al. delivered wavelet-based photograph fusion to easily cognizance at the object with all focal lengths as several vision-related processing tasks can be carried out more effortlessly when wholly substances within the images are bright. In their work Kwok et al. investigated with different datasets, and results show that presented work is extra correct as it does not get suffering from evenness at different activity stages computations.

- **M. Kumar and K. K. Mehta, "A Texture based Tumor detection and automatic Segmentation using Seeded Region Growing Method," International Journal of Computer Technology and Applications, ISSN: 2229-6093, Vol. 2, Issue 4, PP. 855-859 August 2011.**

Kumar and Mehta proposed the texture-based technique in this paper. They highlighted the effects of segmentation if the tumor tissue edges aren't shrill. The performance of the proposed technology may get unwilling results due to those edges. The texture evaluation and seeded region approach turned into executed inside the MATLAB environment.

- **Mahmoud, Dalia & Mohamed, Eltaher. (2012). Brain Tumor Detection Using Artificial Neural Networks. Journal of Science and Technology. 13. 31-39.**

Dalia Mahmoud et al. presented a model using Artificial Neural Networks for tumor detection in brain images. They implemented a computerized recognition system for MR imaging the use of Artificial Neural Networks. That was observed that after the Elman community was used during the recognition system, the period time and the accuracy level were high, in comparison with other ANNs systems. This neural community has a sigmoid characteristic which elevated the extent of accuracy of the tumor segmentation.

- **Marroquin J.L., Vemuri B.C., Botello S., Calderon F. (2002) An Accurate and Efficient Bayesian Method for Automatic Segmentation of Brain MRI. In: Heyden A., Sparr G., Nielsen M., Johansen P. (eds) Computer Vision— ECCV 2002. ECCV 2002. Lecture Notes in Computer Science, vol 2353. Springer, Berlin, Heidelberg.**

L. Marroquin et al. [presented the automated 3d segmentation for brain MRI scans. Using a separate parametric model in preference to a single multiplicative magnificence will lessen the impact on the intensities of a grandeur. Brain atlas is hired to find nonrigid conversion to map the usual brain. This transformation is further used to segment the brain from nonbrain tissues, computing prior probabilities and finding automatic initialization and finally applying the MPM-MAP algorithm to find out optimal segmentation. Major findings from the study show that the MPM-MAP algorithm is comparatively robust than EM in terms of errors while estimating the posterior marginal. For optimal segmentation, the MPM-MAP algorithm involves only the solution of linear systems and is therefore computationally efficient.

- **Minz, Astina, and Chandrakant Mahobiya. "MR Image Classification Using Adaboost for Brain Tumor Type." 2017 IEEE 7th International Advance Computing Conference (IACC) (2017): 701-705.**

Astina minz et al. implemented an operative automatic classification approach for brain image that projected the usage of the AdaBoost gadget mastering algorithm. The proposed system includes three main segments. Pre-processing has eradicated noises in the datasets and converted images into grayscale. Median filtering and thresholding segmentation are implemented in the pre-processed image.

- **Monica Subashini.M, Sarat Kumar Sahoo, "Brain MR Image Segmentation for TumorDetection using Artificial Neural Networks," International Journal of Engineering and Technology (IJET), Vol.5, No 2, Apr-May 2013.**

Monica Subashini and Sarat Kumar Sahoo has suggested a technique for detecting the tumor commencing the brain MR images. They also worked on different techniques, which include pulse-coupled Neural Network and noise removal strategies for reinforcing the mind MRI images and backpropagation network for classifying the brain MRI images from tumor cells. They observed image enhancement and segmentation of the usage of their proposed technique, and the backpropagation network helps in the identification of a tumor in a brain MR image.

- **S. Li, J.T. Kwok, I.W Tsang, and Y. Wang, —Fusing Images with Different Focuses using Support Vector Machines, Proceedings of the IEEE Transaction on Neural Networks, China, November 2007.**

Li et al. report that edge detection, image segmentation, and matching are not easy to achieve in optical lenses that have long focal lengths. Previously, researchers have proposed many techniques for this mechanism, one of which is wavelet-based image fusion. The wavelet function can be improved by applying a discrete wavelet frame transform (DWFT) and a support vector machine (SVM). In this paper, the authors experimented with five sets of 256-level images. Experimental results show that this technique is efficient and more accurate as it does not get affected by consistency verification and activity level measurements. However, the paper is limited to only one task related to fusion, and dynamic ranges are not considered during the calculation.

- **H. Yu and J.L. Fan, ―Three-level Image Segmentation Based on Maximum Fuzzy Partition Entropy of 2-D Histogram and Quantum Genetic Algorithm, Advanced Intelligent Computing Theories, and Applications. With Aspects of Artificial Intelligence. Lecture Notes in Computer Science, Berlin, Heidelberg 2008.**

Yu et al. state that image segmentation is used for extracting meaningful objects from an image. They propose segmenting an image into three parts, including dark, grey and white. Z-function and s-function are used for the fuzzy division of the 2D histogram. Afterward, QGA is used for finding a combination of 12 membership parameters, which have a maximum value. This technique is used to enhance image segmentation and the significance of their work is that three-level image segmentation is used by following the maximum fuzzy partition of 2D Histograms. QGA is selected for the optimal combination of parameters with the fuzzy partition entropy. The proposed method of fuzzy partition entropy of 2D histogram generates better performance than one-dimensional 3-level thresh holding method. Somehow, a large number of possible combinations of 12 parameters in a multi-dimensional fuzzy partition are used, and it is practically not feasible to compute each possible value; therefore, QGA can be used to find the optimal combination.

- **P.S. Mukambika, K Uma Rani, "Segmentation and Classification of MRI Brain Tumor," International Research Journal of Engineering and Technology (IRJET), Vol.4, Issue 7, 2017, pp. 683 – 688, ISSN: 2395-0056**

Mukambika et al. proposed methodology for the subsequent stage's classification of the tumor, whether it is present or not. Their proposed work represents the comparative study of strategies used for tumor identification from MR images, namely the Level set approach and discrete wavelength transforms (DWT) and K-method segmentation algorithms. After that phase, feature extraction is done followed SVM classification.

- **Pan, Yuehao & Huang, Weimin & Lin, Zhiping & Zhu, Wanzheng & Zhou, Jiayin & Wong, Jocelyn & Ding, Zhongxiang. (2015). Brain tumor grading based on Neural Networks and Convolutional Neural Networks. Conference proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference. 2015. 699-702.10.1109/EMBC.2015.7318458.**

Yuehao Pan et al., [16] has used brain MRI pix for getting useful statistics for classifying brain tumor. In their proposed method, they used Convolutional Neural Networks (CNN) algorithms for developing a brain tumor detection system. The performance of their CNN report is measured primarily based on sensitivity and specificity parameters, which have stepped forward when in comparison to the Artificial Neural Networks (ANN).

- **S. Pereira, A. Pinto, V. Alves, and C. A. Silva, "Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images," in IEEE Transactions on Medical Imaging, vol. 35, no. 5, pp. 1240-1251, May 2016.** 12

S. Pereira et al. presented that magnetic resonance prevents physical segmentation time in the medical areas. So, an automatic and reliable segmentation technique for identifying abnormal tissues by using Convolutional Neural Network (CNN) had been proposed in the research work. The massive three-dimensional and underlying roughness amongst brain images makes the process of segmenting the image a severe bissue, so a robust methodology such as CNN is used.

- **S. Roy And S. K. Bandyopadhyay, “Detection and Qualification Of Brain Tumor From MRI Of Brain And Symmetric Analysis,” International Journal Of Information And Communication Technology Research, Volume 2 No.6, June 2012, Pp584-588**

Roy et al. (2012) calculated the tumor affected area for proportioned analysis. They confirmed its software with numerous statistics groups with distinctive tumor sizes, intensities, and location. They showed that their algorithm could robotically hit upon and phase the brain tumor from the given photo. Image pre-processing consists of fleeting that pictures to the filtering technique to remove distractors found in given pictures. They first detect the tumor, segment it and then find out the area of tumor. One of the important aspects is that after performing the quantitative analysis, we can identify the status of an increase in the disease. They have suggested multi-step and modular approach to solve the complex MRI segmentation problem. Tumor detection is the first step in tumor segmentation. They have obtained good results in complex situations. The authors claim that MRI segmentation is one of the essential tasks in the medical area but boring and time-consuming if it is performed manually, so visually study of MRI is more interesting and faster.

- **Sankari Ali, and S. Vigneshwari. "Automatic tumor segmentation using convolutional neural networks." 2017 Third International Conference on Science Technology Engineering & Management (ICONSTEM) (2017): 268-272.**

A. Sankari and S. Vigneshwari has proposed a Convolutional Neural Network (CNN) segmentation, which principally based on the brain tumor classification method. The proposed work used the non-linearity activation feature that's a leaky rectified linear unit (LReLU). They primarily focused on necessary capabilities, which include mean and entropy of the image and analyzed that the CNN algorithm is working higher for representing the complicated and minute capabilities of brain tumor tissues present in the MR Images.

- **T.U Paul and S.K. Bandyopadhyay, ―Segmentation of Brain Tumor from Brain MRI Images Reintroducing K – Means with advanced Dual Localization +MethodTuhin, ‖ International Journal of Engineering Research and Applications, Volume 3, Issue 1, June 2012, ISSN 2278-0882.**

T.U Paul and S.K. Bandyopadhyay has presented the brain segmentation that has automated the use of the Dual Localization technique. In the initial phase, the skull masks are generated for the brain MR images. The tumor areas are improvised using the K-manner procedure. In the final step of their proposed work, they evaluated by its dimensions such as length and breadth.

- **Vaishali et al. (2015) Wavelet-based feature extraction for brain tumor diagnosis—a survey. Int J Res Appl Sci Eng Technol (IJRASET) 3(V), ISSN: 2321-9653**

Vaishali proposed a method that includes step by step procedure starting with image pre-processing followed by extraction of useful objects and finally classification of tumor region. Pre-processing is completed to enhance the image using eliminating the noise via making use of Gaussian filters from the authentic ones. The next step is feature extraction, in which a magnified image is used to extract the feature using a symlet wavelet technique. The very last step is the classification of tumors by the use of a Support vector machine (SVM).

- **Varuna Shree, N., Kumar, T.N.R. Identification and classification of brain tumor MRI images with feature extraction using DWT and probabilistic neural network. Brain Inf. 5, 23–30 (2018) DOI:10.1007/s40708-017-0075-5**

Kumar and Varuna Shree proposed work for the detection tumor region using discrete wavelength transforms (DWT). This work consists of three phases, namely an image enhancement using filtering technique, gray-level co-incidence matrix (GLCM) feature extraction of tumor in addition to DWT based tumor location developing segmentation. It is used to improve overall performance and reduce complexity. The denoised accompanied by the aid of morphological filtering operations which put off the noises that can be even shaped subsequent segmentation technique. The PNN classifier is to use for classifying the abnormality, which is trained by different datasets, and the accuracy is measured within the detection of tumor region of mind MR images.

- **Vinotha, K., 2014. "Brain Tumor Detection and Classification Using Histogram Equalization and Fuzzy Support Vector Machine Approach," International Journal of Engineering and Computer Science ISSN2319- 7242 3(5): 5823-5827.**

K. Vinotha et al. proposed brain tumor detection and the usage of the Histogram Equalization (HE) and the Fuzzy Support Vector Machine (FSVM) classification techniques. The brain MR image is pre-processed with histogram equalization and segmented the apprehensive components from the photo primarily based on the MRF algorithm for the segmentation technique. MRF approach expanded the tumor segmentation accuracy through which the overall performance of the proposed approach changed into advanced.

- **Sing, J.K. & Basu, D.K. & Nasipuri, Mita & Kundu, Megha. (2003). Improved k-means algorithm in the design of RBF neural networks. 2. 841 - 845 Vol.2.**

Sing et al. propose a fuzzy adaptive RRBI-basedneural network for MR brain image segmentation. The hidden layer neuron of FARBF-NN neurons has been fuzzified to reduce nthe oise effect. Basu et al. assert that the medical image segmentation approach involves a combination of texture and boundary information.

The authors maintain that geometric algebra can be used to obtain volumetric data representation using spheres, nonrigid registration of spheres and real-time object tracking. The major contribution of the proposed approach is that the use of the marching cube algorithm reduces the number of primitives to model volumetric data and uses a lesser number of primitives for the registration process, and thus makes the registration process faster. However, the study has employed images obtained from CT scans, which has its own limitations like blurred boundaries and similar grey levels between healthy and non-healthy tissues.

- **Shi, Z., He, L., Suzuki, K., Nakamura, T., & Itoh, H. (2009). Survey on Neural Networks Used for Medical Image Processing. International Journal of computational science, 3(1), 86–100.**

Shi et al. employed neural networks for medical image processing, including the key features of medical image pre-processing, segmentation, and object detection and recognition. The study employed Hopfield and feedforward neural networks. The feedforward and Hopfield neural networks are simple to use and easy to implement. The added advantage of Hopfield neural networks is that it does not require pre-experimental knowledge. The time required to resolve image processing predicament is substantially reduced by using a trained neural network

- **Detection of Tumors in MRI Images Using Artificial Neural Networks**

Automatic defect detection in MR images is very important in many diagnostic and therapeutic applications. This work has introduced one automatic brain tumor detection method to increase the accuracy and yield and decrease the diagnosis time. The goal is classifying the tissues into two classes of normal and abnormal. MR images that have been used here are MR images from normal and abnormal brain tissues. This method uses from neural network to do this classification. The purpose of this project is to classify the brain tissues to normal and abnormal classes automatically, which saves the radiologist time, increases athe ccuracy and yield of diagnosis.

- **Survey on Brain Tumor Detection Techniques Using Magnetic Resonance Images**

The brain tumor is an abnormal growth of cells inside the skull which causes damage to the other cells necessary for functioning human brain. Brain tumor detection is a challenging task due to the complex structure of the human brain. MRI images generated from MRI scanners using strong magnetic fields and radio waves to form images of the body which helps for medical diagnosis. This paper gives an overview of the various techniques used to detect the tumor in the human brain using MRI images.

- **A Neural Network-based Method for Brain Abnormality Detection in MR Images Using Gabor Wavelets**

Nowadays, automatic defects detection in MR images is very important in many diagnostic and therapeutic applications. This paper introduces a Novel automatic brain tumor detection method that uses T1, T2_weighted and PD, MR images to determine any abnormality in brain tissues. Here, it has been tried to give a clear description from brain tissues using Gabor wavelets, energy, entropy, contrast and some other statistic features such as mean, median, variance, correlation, values of maximum and minimum intensity. It is used from a feature selection method to reduce the feature space too. This method uses from neural network to do this classification. The purpose of this project is to classify the brain tissues to normal and abnormal classes automatically, which saves the radiologist time, increases accuracy and yield of diagnosis.

2.1 Problem Statement

The algorithm will result in an extracted image of the tumor from the MR brain image. Brain tumors are to provide aid to clinical diagnosis. The aim is to provide an algorithm. Especially for the medical staff treating the patient. The aim of this work is to define filtering, erosion, dilation, threshold, and outlining of the tumor such as edge detection. A foolproof method of tumor detection in MR brain images. The methods utilized are normally the anatomy of the brain is analyzed by MRI scans or CT scans. The aim of this work is to bring some useful information in simpler form in front of the users, Prove useful for various cases, will provide a better base for the staff to decide. Representation in a simpler form such that it is understandable by everyone. The objective Resultant image will be able to provide information like size, dimension, and position that guarantees the presence of a tumor by combining several procedures to provide. The curing procedure. Finally, we detect whether the given MR brain image has a tumor or not using Convolution Neural Network.

2.2 Existing System

In the first stage, this method includes some noise removal functions improving features that provide better characteristics of medical images for reliable diagnosis using the Balance Contrast Enhancement Technique (BCET). The result of the second stage is subjected to image segmentation using the Fuzzy c-Means (FCM) clustering method. Finally, the Canny edge detection method is applied to detect the fine edges. During the experimental study, we used images containing brain tumors that were characterized by a different location, type of pathology, shape, size, and density, as well as the size of the area of the affected tissue near the tumor space. Detection and extraction of tumors from MRI scan images of the brain are done using MATLAB software. The obtained results demonstrate some resistance to noise. Also, the accuracy of segmentation, in some cases of tumor pathology.

2.3 Disadvantages of the Existing System

By using the fuzzy c-means (FCM) algorithm the accuracy is 80% and the number of iterations is also less. But FCM is highly vulnerable to noise due to not considering the spatial information in image segmentation.

2.4 Proposed System

Here we are going to detect the tumor using a human-trained model we trained our model using images and we got an accuracy of nearly >90% using the CNN algorithm the person can collect their MRI and then upload a copy of an image to our webpage then from there he can get to know whether he is detected with a brain tumor or not and then from there, he can proceed for further treatment.

2.5 Advantages Of the Proposed System

Our system aims to eliminate the necessity of performing a biopsy or surgery to identify whether the image shows any malignancy or not. It is considered the best ml technique for image classification due to its high accuracy. Image pre-processing required is much less compared to other algorithms. It is used over feed-forward neural networks as it can be trained better in the n case of complex images to have higher accuracies. It reduces images to a form that is easier to process without losing features that are critical for a good prediction by applying relevant filters and reusability of weight. It can automatically learn to perform any task just by going through the training data i.e. there is no need for prior knowledge. There is no need for specialized hand-crafted image features like that in the case of SVM, Random Forest, etc.…

3. CHALLENGES IN TUMOR CLASSIFICATION

The identification of tumors is a very challenging task. The location, shape, and structure of tumors vary significantly from patient to patient which makes the segmentation a very challenging task. In the figure shown below, we have shown some images of the same brain slice from different patients, which clearly reflect the variation of the tumor. We can clearly see that the location of the tumor is different in all 8 images/patients shown below. To make it worse, the shape and the intra-tumoral structure are also different for all eight patients/images. In fact, there can be more than one region of the tumor as can be seen from the images.

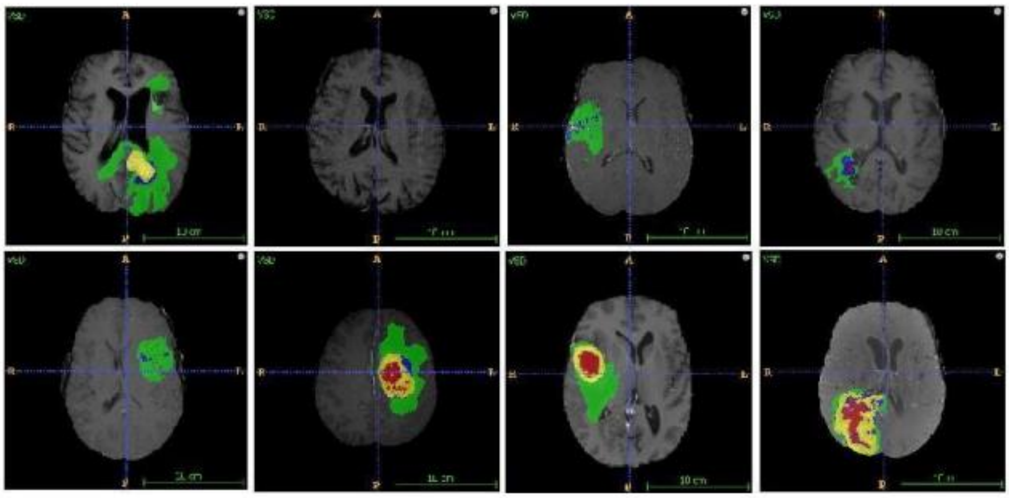


Fig.3.1 Location of tumors in eight different images.

4. SOFTWARE & HARDWARE REQUIREMENTS

4.1 Software Requirements:

Windows: Python 3.6.2 or above, PIP and NumPy 1.13.1

Python:

Python is an interpreted, high-level, general-purpose programming language created by Guido Van Rossum and first released in 1991, Python's design philosophy emphasizes code Readability with its notable use of significant Whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

PIP:

It is the package management system used to install and manage software packages written in Python.

NumPy:

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It contains various features including these important ones :

- A powerful N-dimensional array object.
- Sophisticated (broadcasting) functions.
- Tools for integrating C/C++ and Fortran code.
- Useful linear algebra, Fourier transform, and random number capabilities.

Tensor Flow:

Tensor flow is a free and open-source software library for dataflow and differentiable programming across a range of tasks. It is a symbolic math library, and is also used for machine learning applications such as neural networks. It is used for both research and production at Google.

Keras:

Keras is an open-source neural-network library written in Python. It is capable of running on top of TensorFlow, Microsoft Cognitive Toolkit, R, Theano, or Plaid ML. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. Keras contains numerous implementations of commonly used neural network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code.

OpenCV:

OpenCV (Open source computer vision) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by willow garage and then see (which was later acquired by Intel). The library is cross-platform and free for use under the open-source BSD license. OpenCV supports some models from deep learning frameworks like TensorFlow, Torch, PyTorch (after converting to an ONNX model), and Caffe according to a defined list of supported layers. It promotes Open Vision Capsules.

4.2 Hardware Requirements:

 Processor: Intel core i5 or above.

 64-bit, quad-core, 2.5 GHz minimum per core

 Ram: 4 GB or more

 Hard disk: 10 GB of available space or more.

 Display: Dual XGA (1024 x 768) or higher resolution monitors

 Operating system: Windows

5. METHODOLOGY

5.1 Machine learning Life cycle

Machine learning has given computer systems the ability to automatically learn without being explicitly programmed. But how does a machine learning system work? So, it can be described using the life cycle of machine learning. The machine learning life cycle is a cyclic process to build an efficient machine learning project. The main purpose of the life cycle is to find a solution to the problem or project.

The machine learning life cycle involves seven major steps, which are given below:

- **Gathering Data**
- **Data Preparation**
- **Data Wrangling**
- **Analyse Data**
- **Train the model**
- **Test the model**
- **Deployment**

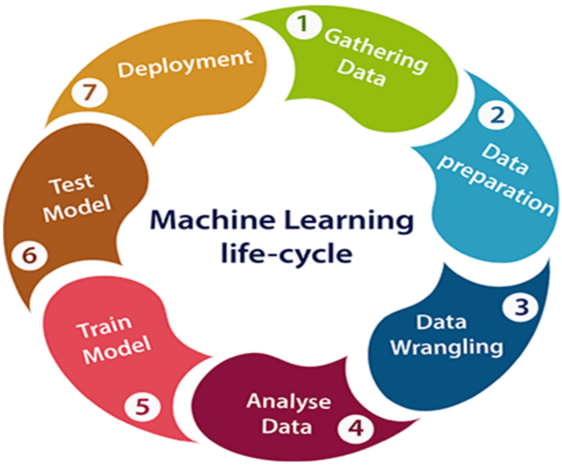


Figure 5.1 Machine learning life cycle

The most important thing in the complete process is to understand the problem and to know the purpose of the problem. Therefore, before starting the life cycle, we need to understand the problem because the good result depends on a better understanding of the problem.

5.2 Architecture:

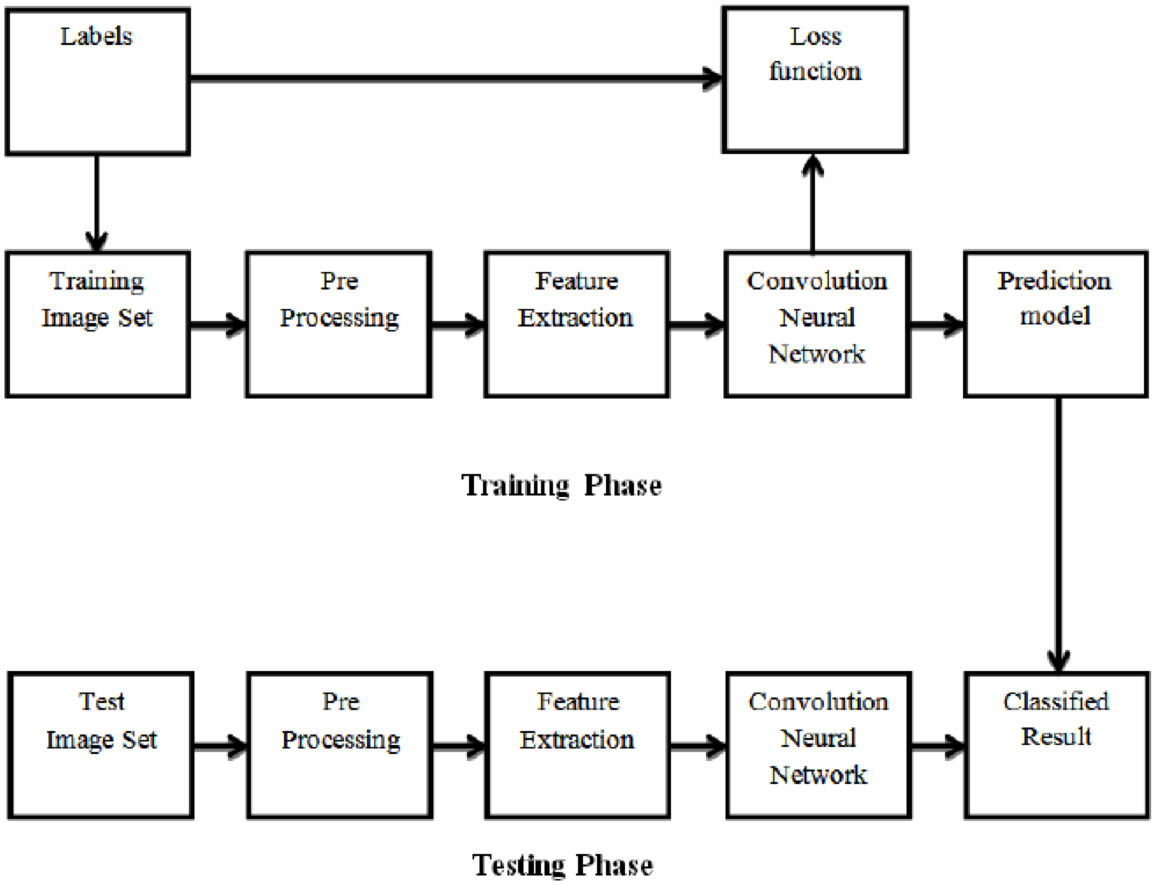


Figure 5.2 Architecture

5.3.1 Gathering Data:

Data Gathering is the first step of the machine learning life cycle. The goal of this step is to identify and obtain all data-related problems.

In this step, we need to identify the different data sources, as data can be collected from various sources such as files, database, internet, or mobile devices. It is one of the most important steps of the life cycle. The quantity and quality of the collected data will determine the efficiency of the output. The more data, the more accurate the prediction will be.

This step includes the below tasks:

- Identify various data sources
- Collect data
- Integrate the data obtained from different sources

By performing the above task, we get a coherent set of data, also called a **dataset**.

It will be used in further steps.

5.4.2 Data preparation

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.

In this step, first, we put all data together, and then randomize the ordering of data.

This step can be further divided into two processes:

- A better understanding of data leads to an effective outcome. In this, we find Correlations, general trends, and outliers.
- Data pre-processing:

  Now the next step is pre-processing of data for its analysis.

5.3.3 Data Wrangling

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:

- Missing Values
- Duplicate data
- Invalid data
- Noise

So, we use various filtering techniques to clean the data.

It is mandatory to detect and remove the above issues because it can negatively affect the quality of the outcome.

5.3.4 Data Analysis

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- Selection of analytical techniques
- Building models
- Review the result

The aim of this step is to build a machine learning model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as **Classification**, **Regression**, **Cluster analysis**, **Association**, etc. then build the model using prepared data, and evaluate the model.

Hence, in this step, we take the data and use machine learning algorithms to build the model.

5.3.5 Train Model

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.

We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and features.

5.3.6 Test Model

Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.

Testing the model determines the percentage accuracy of the model as per the requirement of the project or problem.

5.3.7 Deployment

The last step of the machine learning life cycle is deployment, where we deploy the model in a real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

6. Dataset

A dataset is **a collection of data curated for a machine-learning project**. An image dataset includes digital images curated for testing, training, and evaluating the performance of machine learning and artificial intelligence (AI) algorithms, commonly computer vision algorithms.

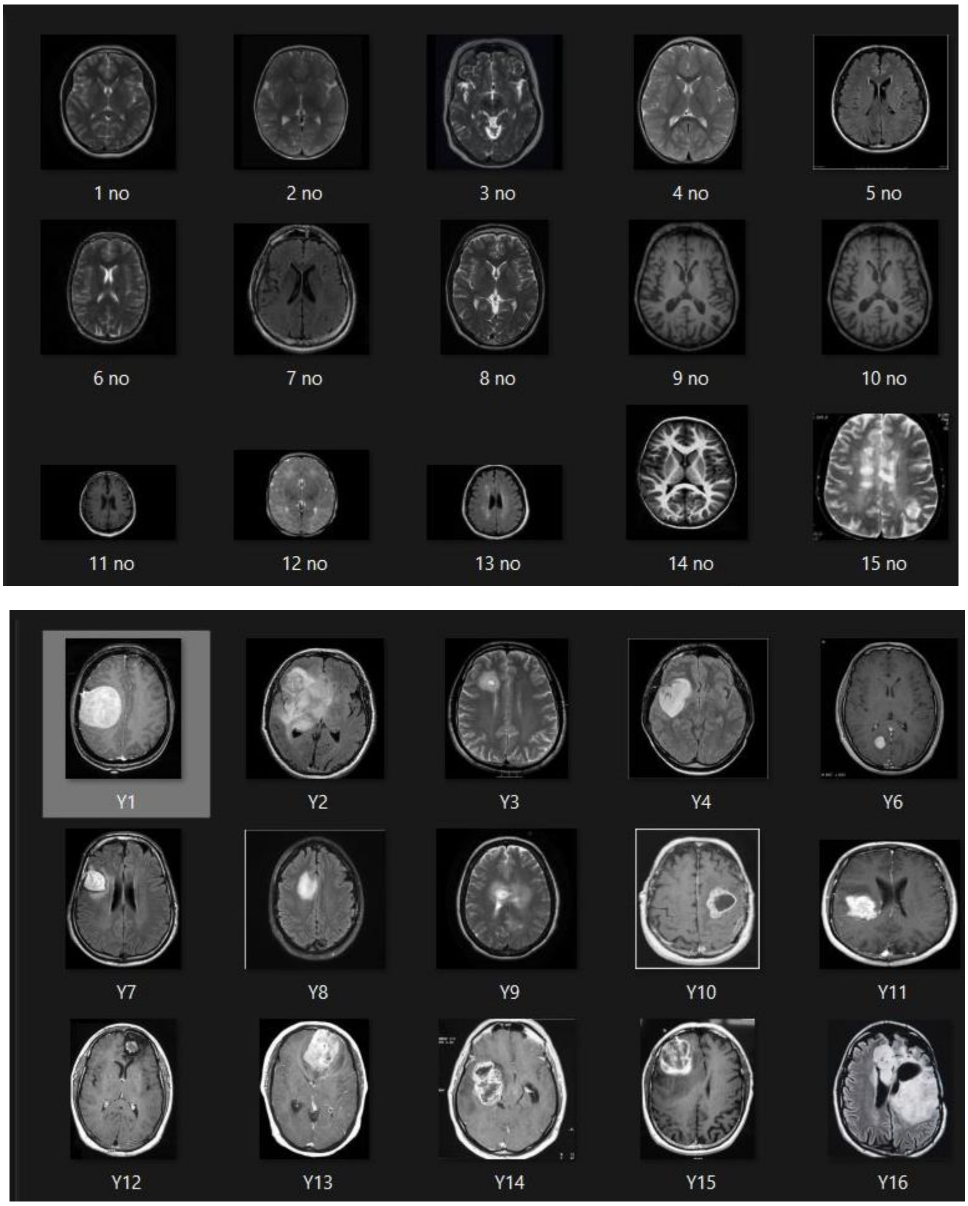


Figure 6.1 Dataset

6.1 Need for Dataset

To work with machine learning projects, we need a huge amount of data, because, without the data, one cannot train ML/AI models. Collecting and preparing the dataset is one of the most crucial parts while creating an ML/AI project.

The technology applied to any ML project cannot work properly if the dataset is not well prepared and pre-processed.

During the development of the ML project, the developers completely rely on the datasets. In building ML applications, datasets are divided into two parts:

- Training dataset
- Test Dataset

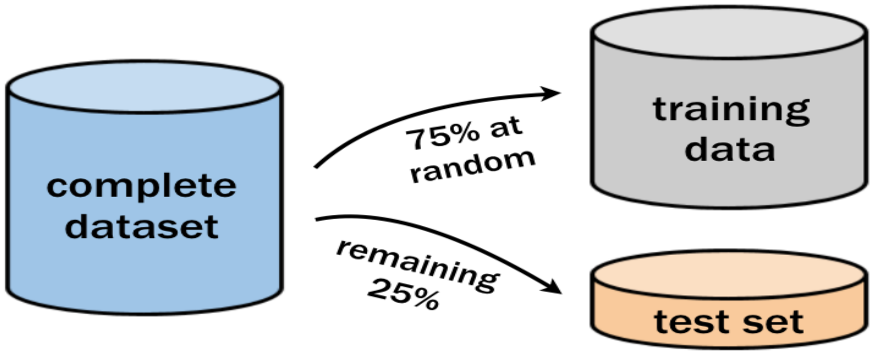


Figure 6.2 Dataset Division into training and testing sets

6.2 Image Pre-processing

In medical image processing, medical images are corrupted by different types of noises. It is very important to obtain precise images to facilitate accurate observations for the given application [78]. Pre-processing steps include filtering, morphological operations, etc. • Filtering: Image filtering is useful for many applications, including smoothing, sharpening, removing noise, and edge detection. A filter is defined by a kernel, which is a small array applied to each pixel and its neighbors within an image. The process used to apply filters to an image is known as convolution and may be applied in either the spatial or frequency domain. Spatial domain filtering can be classified into two types smoothing and sharpening filters, according to their outputs. – Smoothening Filter: Smoothening spatial filters are used for noise reductions and blurring operations. It includes box filter, Gaussian blur filter, median blur filter, bilateral filter, etc. 1. Box Filtering is basically an average-of-surrounding-pixel kind of image filtering. 2. Gaussian blur filtering is a 2D convolution operator that is used to 'blur' images and remove detail and noise. In this sense, it is similar to the mean filter, but it uses a different kernel that represents the shape of a Gaussian ('bell-shaped') hump. The degree of smoothing is determined by the standard deviation of the Gaussian. 3. Median blur filter is normally used to reduce noise in an image. Instead of simply replacing the pixel value with the mean of neighbouring pixel values, it replaces it with the median of those values. 4. Bilateral filter is a non-linear, edge-preserving, and noise-reducing smoothing filter for images. This preserves sharp edges. – Sharpening Filter: Sharpening spatial filters seek to highlight fine details. It removes blurring from images and highlights edges. Sharpening filters are based on spatial differentiation. It includes the Laplacian filter, Sobel filter, difference filter, etc. 1. Laplacian filters are derivative filters used to find areas of rapid change (edges) in images. 2. Sobel filters are typically used for edge detection. 3. Difference filters enhance the details in the direction specific to the mask selected.

Morphological Operations: Morphology means pixel shape-based analysis. The objective of using morphological operations is to remove the imperfections in the structure of the image [79]. The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. Morphological operations also include opening, closing, hit-and-miss transform, etc. Morphological erosion removes islands and small objects so that only substantive objects remain.

$$I \oplus H = \{(p + q) \mid f\ or\ ever\ y\ p \in I, q \in H\}$$

where I is the original image and H is the structuring element. Dilation is the opposite of Erosion. The value of the output pixel is the maximum value of all pixels in the neighbourhood. In a binary image, a pixel is set to 1 if any of the neighbouring pixels have the value 1. Morphological dilation makes objects more visible and fills in small holes in objects.

$$I\ H = \{p \in Z\ 2\ |(p + q) \in I, f\ or\ ever\ y\ q \in H\}$$

where I is the original image and H is the structuring element.

In brain tumor segmentation, morphological operations are needed for removing soft tissue boundaries and for effective segmentation of the tumor portion.

Segmentation:

Segmentation means dividing the image into different regions and separating objects from the background. Accurate segmentation of objects of interest in an image greatly facilitates further analysis of these objects. There are various types of segmentation algorithms such as edge-based, thresholding-based, region-based, clustering based etc.

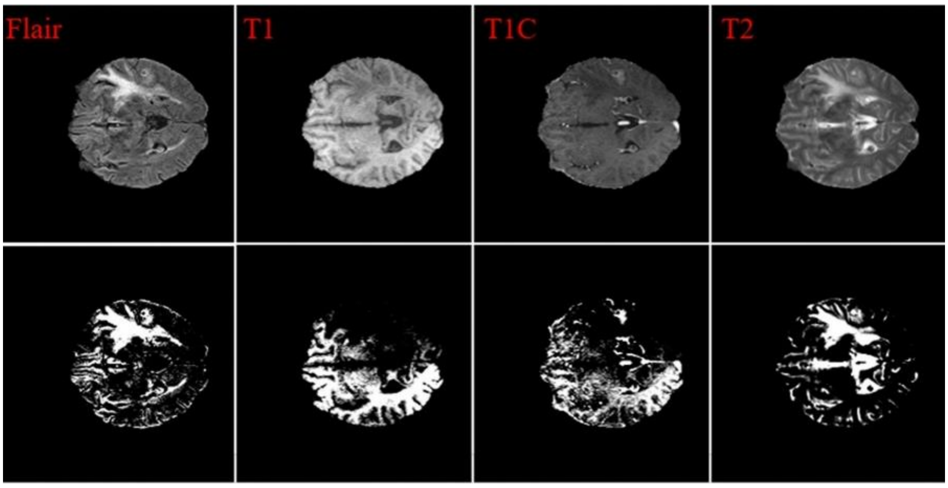


Figure 6.3 Image Segmentation

6.3 K-Means Clustering:

K-Means clustering is a type of unsupervised learning. So, when we have unlabelled data, we can use K-Means Clustering. The K-means clustering algorithm is used to find groups that have not been explicitly labelled in the data. The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K [82]. Based on the features that are provided, this algorithm iteratively assigns each data point to one of the K groups. These are the results that the K-Means Clustering algorithm gives: the centroids of the K clusters, which can be used to label new data and labels for the training data (each data point is assigned to a single cluster). A cluster refers to a collection of data points aggregated together because of certain similarities. A centroid is an imaginary or real location representing the centre of the cluster. Each centroid of a cluster is a collection of feature values that define the resulting groups. Examining the centroid feature weights can be used to qualitatively interpret what kind of group each cluster represents. Figure 4.3 is the visual representation of the segmentation which was done by K-means clustering technique.

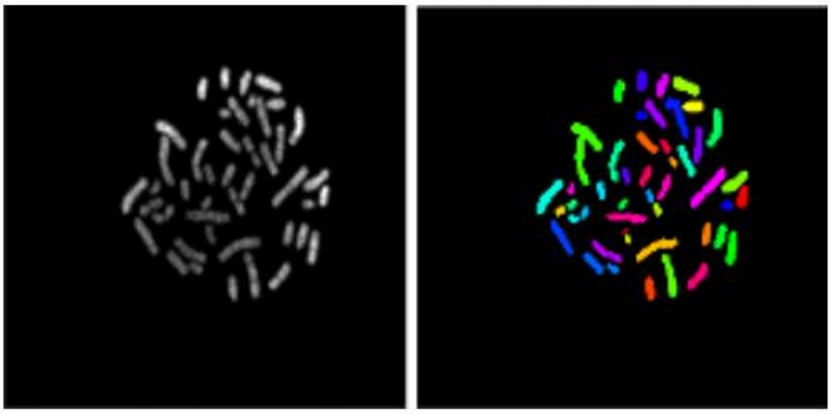


Figure 6.4 Segmentation Using K-Means Clustering

7. DEEP LEARNING

Deep Learning is a subset of machine learning algorithms that is very good at recognizing patterns but typically requires a large amount of data. Deep learning excels in recognizing objects in images as it is implemented using three or more layers of artificial neural networks where each layer is responsible for extracting one or more features of the image.

7.1 Neuron

The basic building unit of neural networks is artificial neurons, which imitate human brain neurons. These artificial neurons are powerful computational units that have weighted input signals and produce an output signal using an activation function. These neurons are spread across the several layers in a neural network.

Figure 7.1 depicts the information of a neuron: A neuron has three parameters, namely:

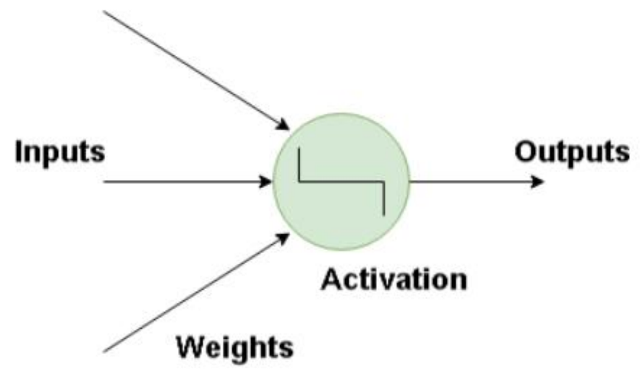


Figure 7.1 Basic structure of a neuron

- Weight: When a signal (value) arrives, a neuron gets multiplied by a weight value. If a neuron has three inputs, it has three weight values which can be adjusted during training time.

- Bias: It is an extra input to neurons and it is always 1, and has its own connection weight. This makes sure that even when all the inputs are none (all 0's) there is going to be activation.

- Activation Function: Activation functions are also known as transfer functions. It helps in classification or partition. It uses a threshold, and according to this threshold, we make the division into two or many classes.

7.2 Artificial Neural Network

Deep learning consists of artificial neural networks that are modelled on similar networks present in the human brain. These neural networks work in multiple layers so this kind of machine learning is called deep learning.

Artificial neural networks are a way of calculating an output from an input (a classification) using weighted connections ("synapses") that are calculated from repeated iterations through training data. Each pass through the training data alters the weights such that the neural network produces the output with greater "accuracy" (lower error rate). The combination of working memory and speed is crucial when we're doing hundreds of thousands of matrix multiplications. Figure 4.6 represents a simple artificial neural network:

An artificial neural network generally has three layers. Layers are made up of some nodes which are interconnected. The three layers of ANN are described in the following:

• Input Layer: This layer consists of neurons and they just receive the inputs and pass it to the next layer. The number of layers in the input layer should be equal to the attributes or features in the dataset.

• Hidden Layer: In between input and output layer there are hidden layers based on the type of model. Hidden layers contain vast number of neurons. The neurons in the hidden layer apply transformations to the inputs before passing them to the next layer. As the network is trained the weights get updated, to be more predictive. The actual processing of the data is done via a system of weighted connections in the hidden layer. The hidden layers are linked to the output layer.

• Output Layer: The output layer is the predicted feature, or class in a classification problem, it basically depends in the type of the built model. The output layer gives the output based on the information passed from the hidden layer.

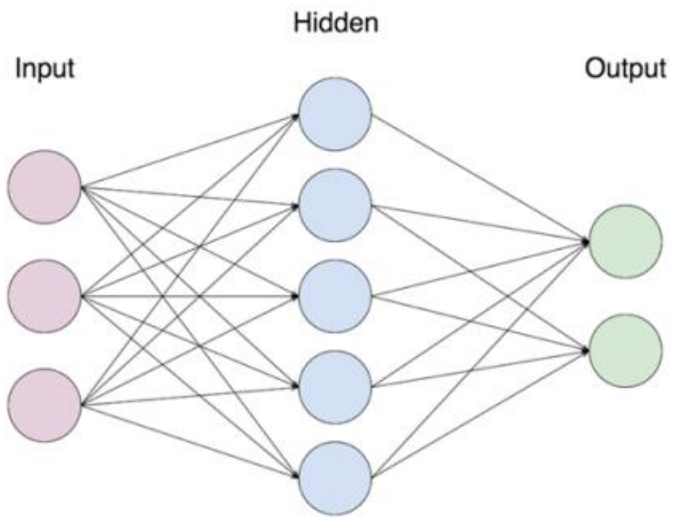


Figure 7.2 A simple Artificial Neural Network

The internal structure of ANN can be changed by itself based on the information passing through it. This is done by the adjustment of the weights. Every connection in the neural network generally has a weight that controls the signal between the two neurons. If the output is good, the adjustment is no longer needed, but if the output is poor, then the system adapts by changing the weights to improve the output. The performance evaluation of the output is done by the system by comparing the output with the original output given before in the training mode.

7.3 Using Neural Networks for Images

A neural network can be used to recognize or detect object categories but it will require more work to uniquely identify an object. A classical neural network requires inputting a set of features extracted from each of the images. A deep neural network (DNN) works with image pixels.

An image can be represented as a matrix, each element of the matrix containing color in formation for a pixel. The matrix is used as input data into the neural network. The small dimensions of the images, easily and quickly help learn and establish the size of the vector and the number of input vectors. The transfer function used is also called an activation function. Processing of images with artificial neural networks involves different processes, such as:

- Image pre-processing: It is an operation that shows a picture (contrast enhancement, noise reduction) with the same dimensions as the original image. The objective of images' pre-processing with ANN consists in improving, restoring, or rebuilding im- ages.

- Data reduction or feature extraction: This step involves extracting a number of features smaller than the number of pixels in the input window. The operation consists in compressing the image followed by extracting geometric characteristics such as edges, corners, joints, facial features, etc.

- Segmentation: Segmentation means the division of an image into different meaningful regions.

- Recognition: It involves the determination of objects in an image and their classification.

Processing images with artificial neural networks successfully resolves the problems of classification, identification, authentication, diagnostics, optimization, and approximation.

7.4 Types of Neural Networks

There are several kinds of artificial neural networks. These types of networks are implemented based on mathematical operations and a set of parameters required to determine the output. Some of the popular neural networks are:

- Feedforward Neural Network: This neural network is one of the simplest forms of ANN, where the data or the input travels in one direction. The data passes through the input nodes and exits on the output nodes. This neural network may or may not have hidden layers.

In simple words, it has a front-propagated wave and no backpropagation by using a classifying activation function usually.

- Radial Basis Function Neural Network: Radial basis functions consider the distance of a point with respect to the centre. RBF functions have two layers, first where the features are combined with the Radial Basis Function in the inner layer and then the output of these features is taken into consideration while computing the same output in the next time step which is basically a memory.

- Korhonen Self-Organizing Neural Network: A self-organizing map (SOM) is a type of artificial neural network (ANN) that is trained using unsupervised learning to pro- duce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map, and is, therefore, a method to do dimensionality reduction.

- Recurrent Neural Network: The Recurrent Neural Network works on the principle of saving the output of a layer and feeding this back to the input to help in predicting the outcome of the layer. This is also known as Long Short-Term Memory.

- Convolutional Neural Network: Convolutional neural networks are similar to feed-forward neural networks, where the neurons have learnable weights and biases.

- Modular Neural Network: Modular Neural Networks have a collection of different networks working independently and contributing towards the output. Each neural network has a set of inputs that are unique compared to other networks constructing and performing sub-tasks. These networks do not interact or signal each other in accomplishing tasks. Among these mostly used neural networks, Convolutional Neural Networks are applied in techniques like signal processing and image classification techniques.

7.5 Convolutional Neural Network (CNN)

The basic idea of a Convolutional Neural Network was introduced by Kunihiko Fukushima in the 1980s [88]. Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification. Computer vision techniques are dominated by convolutional neural networks because of their accuracy in image classification. CNN is a class of deep, feed-forward artificial neural networks (where connections between nodes do not form a cycle) & uses a variation of multi-layer perceptron designed to require minimal pre-processing.

ConvNet architectures make the explicit assumption that the inputs are images, which allows us to encode certain properties into the architecture. These then make the forward function more efficient to implement and vastly reduce the number of parameters in the network.

Why CNN is different from simple Neural Networks: Convolutional Neural Networks have a different architecture than regular Neural Networks [86]. Regular Neural Networks transform an input by putting it through a series of hidden layers. Every layer is made up of a set of neurons, where each layer is fully connected to all neurons in the layer before and where neurons in a single layer function completely independently and do not share any connections between themselves. Finally, there is a last fully-connected layer, which is the output layer that represents the predictions. Regular Neural Networks do not scale well to full images.

Convolutional Neural Networks are a bit different. First of all, the layers are organized in three dimensions: width, height, and depth. Further, the neurons in one layer do not connect to all the neurons in the next layer but only to a small region of it. Lastly, the final output will be reduced to a single vector of probability scores, organized along the depth dimension.

Moreover, CNN’s perform convolution operations in case of matrix multiplication

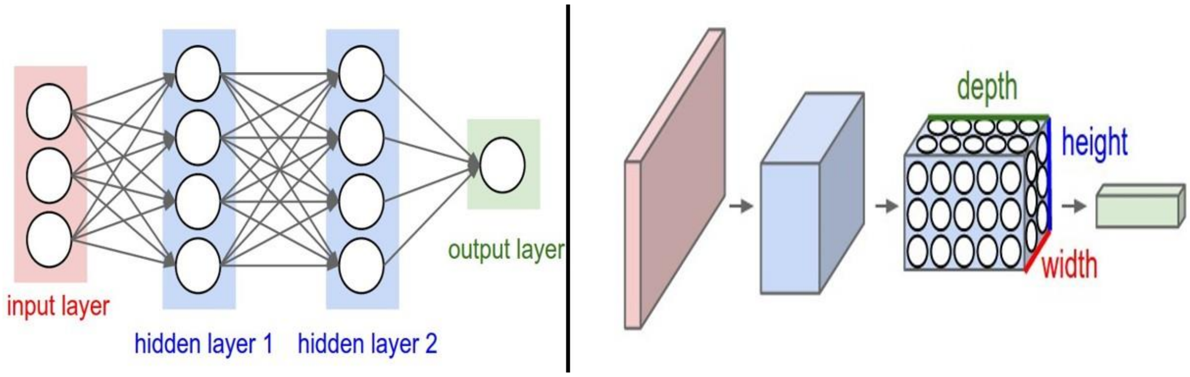


Figure 7.3 Hidden layers in neural networks

In the above, in figure 4.7, the left side represents a regular three Layer neural network. On the other hand, the right side of the figure represents a CNN that arranges its neurons in three dimensions (width, height, and depth).

Every layer of a CNN transforms the 3D input volume to a 3D output volume of neuron activations. In this example, the red input layer holds the image, so its width and height would be the dimensions of the image, and the depth would be three (Red, Green, Blue channels).

The Convolution Operation: Convolutional Neural Networks perform a mathematical operation, known as convolution operation. Convolution is a mathematical operation on two functions (f and g) and it produces a third function. The convolution operation of f and g is denoted as f*g. It is defined as the integral of the product of the two functions after

one is reversed and shifted. This operation is a particular kind of integral transform

$(f \square g) t) = f(r)g(t - r)\, dr$

There are three elements that enter into the convolution operation

Input image: It is the image that is given as input.

Feature detector: The feature detector is often referred to as a “kernel" or a “ filter". Sometimes a 5*5 or a 7*7 matrix is used as a feature detector.

Feature map: The feature map is also known as an activation map. It is called a feature map because it is also a mapping of where a certain kind of feature is found in the image.

Figure 7.4: represents the convolution operation.

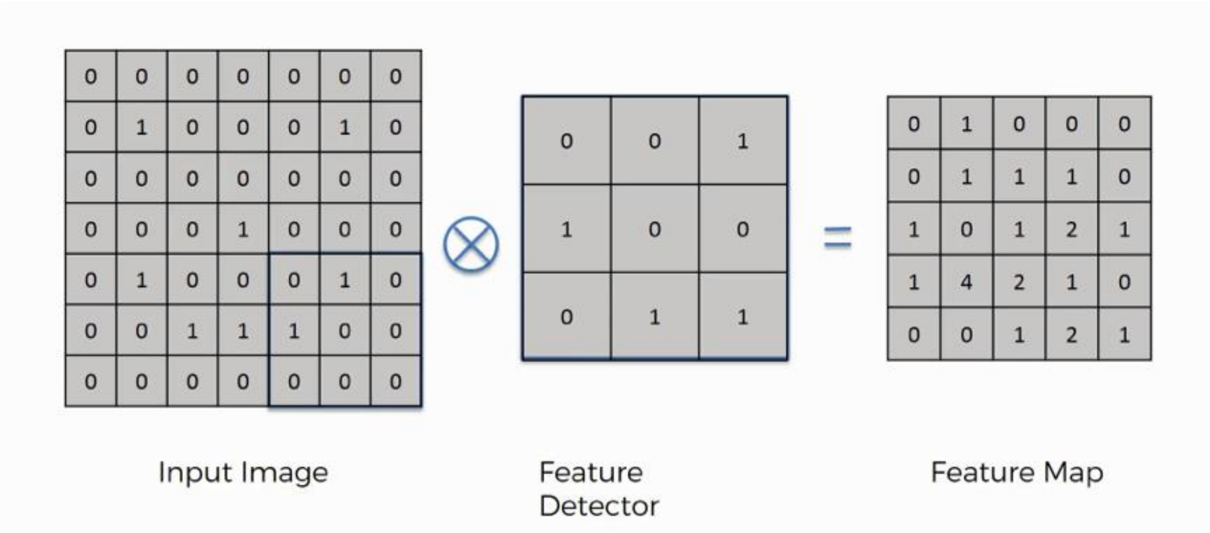


Figure 7.4 represents the convolution operation.

(iv) The Convolution Arithmetic: Here the facts are shown how properties of an output image change from input by some factors [90].

Let, Input Image Size = I, (means width I and Height I i.e. image size is I * I) Filter Size = F, (Filter is F * F)

Number of Filters = K Number of Strides = S Amount of Zero Padding = P

Then the output image size will be:

$O = I - F + 2P + 1$

Forward Pass for Convolutional Layer: In the forward pass of the convolutional layer, a filter performs dot multiplication with all the parts of the input matrix one after another of the filter size, then sums all the elements of the single dot product and adds the bias value with it and places the final value in the corresponding location of the output matrix for that dot product. This process continues across the full input image with all filters and for each filter there comes an output.

Back-propagation for Convolutional Layer: In back-propagation, the cost function is first found out, then this cost function measures the displacement with the output. After that, applying gradient descent on this function will update the filter value of the previous layer. This process continues until it reaches to the input layer.

The iteration of the forward pass and back-propagation will continue until the network finds the desired output.

Layers Used to a Build CNN Model: A simple CNN is a sequence of layers, and every layer of a CNN transforms one volume of activation to another through a differentiable function. Three main types of layers used to build CNN architectures are:

Convolutional Layer:

The Convolutional layer is the core block of the Convolutional Neural Network. It has some special properties. It does most of the computational heavy lifting. The CONV layer's parameters consist of a set of learnable filters. Every filter is small spatially (along width and height), but extends through the full depth of the input volume.

For example, a typical 3X3 filter on a first layer of a ConvNet might have size 5*5*3 (i.e. 5 pixels width and height, and 3 because images have depth 3, the color channels). During the forward pass, each filter is convolved across the width and height of the input volume, and dot products are computed between the entries of the filter and the input at any position. As the filters are slid over the width and height of the thein-putt volume, a 2-dimensional activation map will be produced that gives the responses of that filter at every spatial position. Intuitively, the network will learn filters that ac- activate when they see some type of visual feature such as an edge of some orientation. These activation maps are stacked along the depth dimension and the output volume is produced.

Figure 4.9 shows an example of convolution in the convolutional layer of CNN. Moreover, the Convolutional layer has some basic features [90], such as:

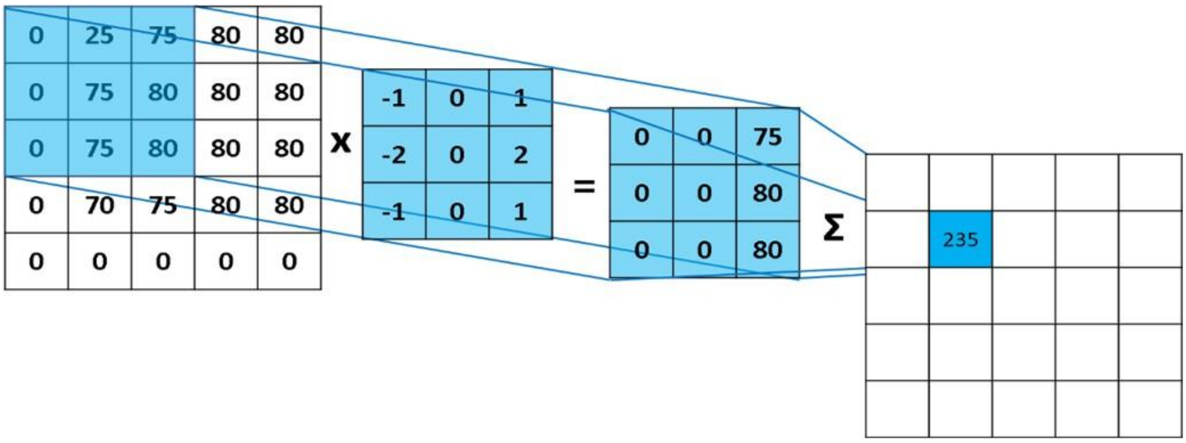


Figure 7.5 Convolution Operation of CNN

Parameter Sharing: Parameter sharing is sharing of weights by all neurons in a particular feature map.

Local Connectivity: Local connectivity is the concept of each neural connected only to a subset of the input image (unlike a neural network where all the neurons are fully connected)

These features help to reduce the number of parameters in the whole system and makes the computation more efficient.

Three hyper-parameters control the size of the output volume of the convolutional layer. These parameters are:

Depth:

The depth of input volume in first layer is the number of color channels of that input image. If the input image is a color image then the depth is 3 that is the red channel, the green channel and the blue channel. If the image is black and white or gray-scale, then the depth is 1.

The depth of the output volume is the number of filters that we use in the input.

Figure 4.8 represents the depth which changes from 3 to 32 when we use 32 filters and the image size gets smaller.

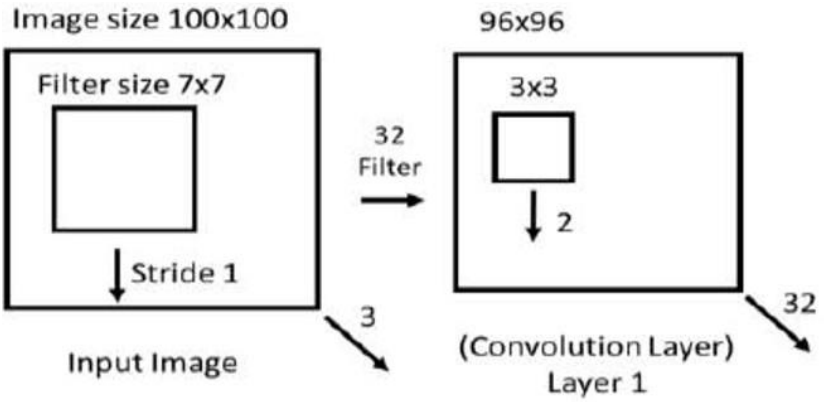


Figure 7.6 Depth changing from 3 to 32 using 32 filters–

Stride:

Stride is used to slide along with dth and height of the input image. When the stride is 1, then we move the filters one pixel at a time. When the stride is 2, then the filters jump 2 pixels at a time as we slide them around.

Figure 4.11: Sliding filter along an input image when the stride is 1

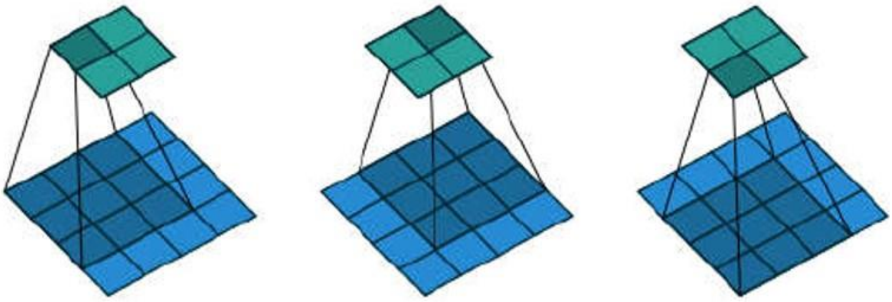


Figure7.7 Shows that a 2*2 filter moves along the input size of 4*4 through width and height when the stride is 1.

Zero Padding:

Sometimes in the input layer, we pad the input image with zero which

is called zero padding. Zero padding allows us to control the size of the input layer. If we don't use zero-padding, sometimes some property from the edges can be lost.

Figure 4.12 illustrates the scenario of zero-padding of an input.

.

Pooling Layer:

The pooling layer is another building block of CNN. Usually pooling layer is placed after the convolutional layer. Its function is to progressively reduce the spatial size of the representation to reduce the number of parameters and computation in the network, and hence to also control overfitting. The Pooling Layer operates independently on every depth slice of the input and resizes it spatially. Pooling does not affect the depth dimension of the input volume.

0	0	0	0	0	0
0	35	19	25	6	0
0	13	22	16	53	0
0	4	3	7	10	0
0	9	8	1	3	0
0	0	0	0	0	0

Figure7.8 Zero-padding of an input (padding amount = 1)

The task of pooling is done by summarizing the sub-regions of the input using some methods like taking the average or the maximum or the minimum value of the sub-regions only. These methods are called pooling functions.

Different Kinds of Pooling Functions: The pooling layer consists of some sym- metric aggregation functions such as:

Max Pooling: It returns the maximum value from its rectangular neighborhood.

Average Pooling: It returns the maximum value from its rectangular neigh- borhood.

Weighted Average Pooling: It calculates its neighborhood weight based on distance from its center pixel.

L2 Norm Pooling: It returns the square root sum of its rectangular neighborhood. In most of the ConvNet architectures, Max Pooling is used to reduce computational cost.

Pooling Layer Arithmetic: The pooling layer works by sliding the window or filter across the input.

Let, Spatial Extent = f

Stride=s window size =w

The equation [90] shows that the output size from the pooling layer will be,

O = , w − f , +s

Fully-Connected Layer:

In a fully connected layer, every neuron is connected to its previous layer neuron like the neural network. Its activation is also computed by matrix multiplication with its weight followed by bias as like neural network. Usually, fully connected layer is a column vector.

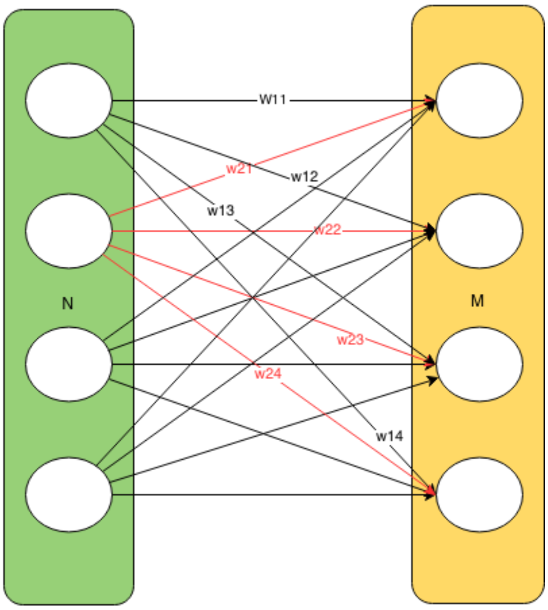


Figure7.9 Fully Connected Layer of CNN

Figure 7.9: shows the connection between two layers where the right one is the fully connected layer.

Activation Function:

Activation functions are used to introduce non-linearity to neural networks. It squashes the values in a smaller range. For example, a sigmoid activation function squashes values between a ranges 0 to 1.

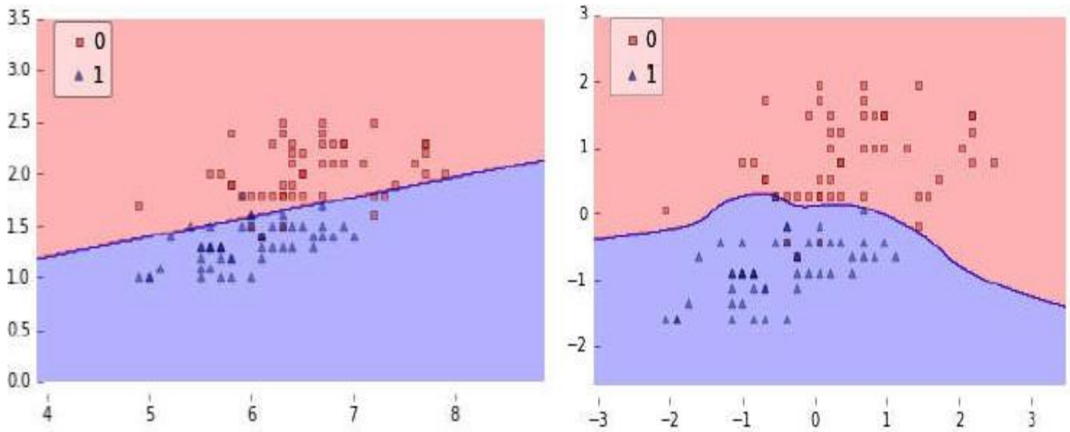


Figure 7.10 Before(left) and after(right) applying activation function

The scenario of applying logistic regression without the application of the activation function and the figure on the right side is the same scenario with the application of the activation function.

Commonly Used Activation Functions:

There are many activation functions used in the deep learning industry. Here we will discuss some activation functions in brief which are commonly in use.

Sigmoid:

The Sigmoid function bounds the input value in between the 0 to 1 range. For a large positive number, it returns 1 and for a large negative number, it returns

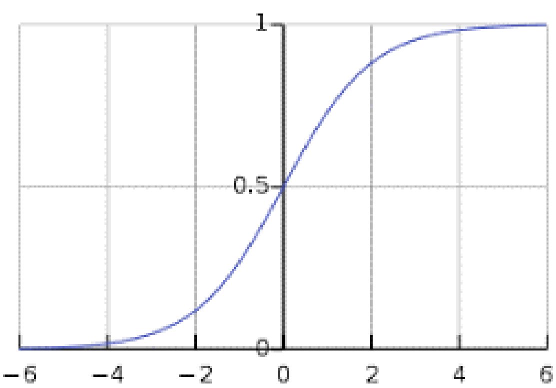


Figure 7.11: Curve of Sigmoid Function

Advantages of Sigmoid Function:

It Provides a smooth gradient and prevents any ‘jump’ in output values. 2. It normalizes the output of each neuron.3.It enables clear predictions.

Disadvantages of Sigmoid Function:

It causes a ‘vanishing gradient’ problem. For very high or very low input values, there is almost no change to the prediction. This can result in the network refusing to learn further, or being too slow to reach an accurate prediction.

Outputs are not zero-centered.

Tanh /Hyperbolic Tangent:

This function is like the sigmoid function. It bounds all real numbers to the range [-1, 1]. The tanh function is mainly used classification between two classes [91].

Figure 4.16 represents the curve of the hyperbolic tangent function:

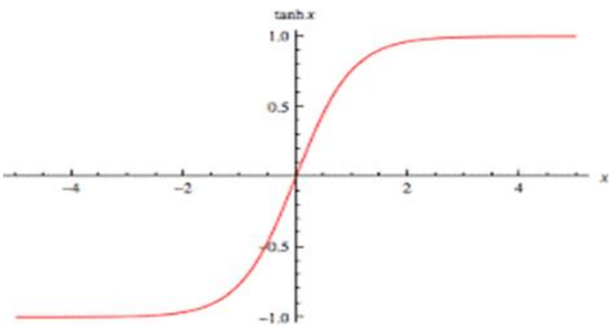


Figure 7.12 Curve of tanh Function

Advantages of the Function:

This function is zero-centered, which makes it easier to model inputs that have strongly negative, neutral, and strongly positive values.

Otherwise like the sigmoid function.

Disadvantages of tanh Function:

The disadvantages of tanh function are as like as the sigmoid function.

ReLU (Rectified Linear Unit)

ReLU refers to Rectified Linear Unit. It simply thresholds the input value to zero. For positive, it returns the number and for negative, it returns 0. In AlexNet architecture, after using ReLU as an activation function, it was 6 times faster than using the tanh function [91]. The formula of ReLu is as follows

f (x) = max (0, x)

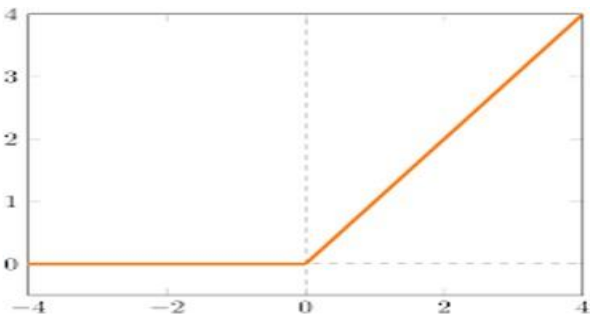


Figure 7.13 Curve of ReLU Function

Advantages of ReLU Function:

- It is computationally efficient and allows the network to converge very quickly.
- Although it looks like a linear function, ReLU is non-linear.
- It has a derivative function and allows for back-propagation.

Disadvantages of ReLU Function:

It introduces the Dying ReLU problem. When inputs approach zero or are negative, the gradient of the function becomes zero, the network cannot perform back-propagation and cannot learn

Regularization Function:

In machine learning, regularization is a way to prevent over-fitting. Regularization reduces over-fitting by adding a penalty to the loss function. By adding this penalty, the model is trained such that it does not learn an interdependent set of feature weights.

If a simple model is not performing well at predicting due to poor generalization and a complex model may not perform well due to over-fitting. In this case, regularization helps to choose the preferred complexity for the model.

The two most important regularization techniques are Dropout and Batch Normalization.

Dropout:

A dropout is an approach to regularization in neural networks which helps reduce interdependent learning amongst the neurons. Dropout forces a neural network to learn more robust features that are useful in conjunction with many different random subsets of the other neurons.

During the forward pass, Dropout temporarily removes some of the neurons. For example, if we fix dropout to 20%, then every 1 out of 5 neurons will be inactivated during the forward pass. During the backward pass, any weight update will not be applied to these neurons.

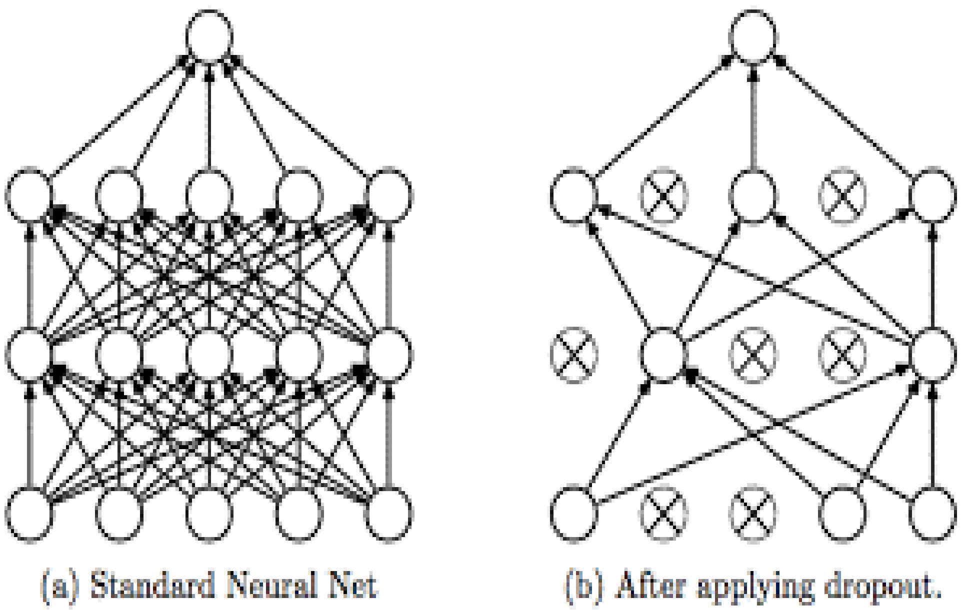


Figure 7.14 Network before and after

Batch Normalization:

Batch Normalization is a method to reduce the internal co-variate shift in neural networks, leading to the possible usage of higher learning rates. In principle, the method adds an additional step between the layers, in which the output of the layer before is normalized. BN further prevents smaller changes to the parameters to amplify and thereby allows higher learning rates, making the network even faster.

Batch normalization (BN) consists of two algorithms. The first algorithm is the transformation of the original input of a layer x to the shifted and normalized value y. The second algorithm is the overall training of a batch-normalized network.

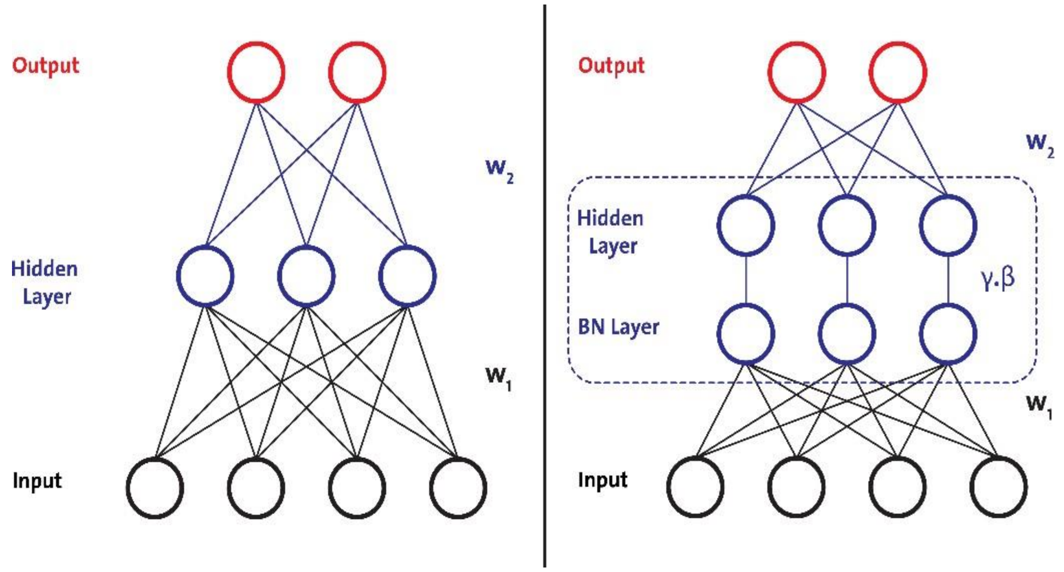


Figure 7.15 A Neural Network before and after Batch Normalizatio

Hyper-parameters of CNN architecture:

In our proposed model, we used different types of hyper-parameters. Following is a brief discussion about hyper-parameters.

Bias:

Bias is an error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs (under-fitting).

glorot_uniform: It is named Xavier normal initializer. It draws samples from a uniform distribution within [-limit, limit].

Beta_1 and Beta_2:

The algorithm calculates an exponential moving average of the gradient and the squared gradient, and the parameters beta1 and beta2 control the decay rates of these moving averages.

Epsilon:

It is a very small number to prevent any division by zero in the implementation.

Dropout or Decay weight: A simple and powerful regularization technique for neural networks and deep learning models is a dropout.

Amstrad:

AMSGrad (in Keras, it is controlled by setting AMS grad = True for Adam optimizer), which uses the maximum of past squared gradients in order to allow the rarely-occurring minibatches with large and informative gradients to have greater in- fluence on the overall direction, otherwise diminished by exponential averaging in plain Adam.

Epoch:

Epoch is defined as the number forward and backward pass of all training examples.

Batch Size: The number of training examples in one forward/backward pass.

Learning rate :

The learning rate or step size in machine learning is a hyper-parameter that determines to what extent newly acquired information overrides old information.

Limitations of CNN :

CNN only captures local 'spatial' patterns in data. If the data can't be made to look like an image, ConvNets are less useful. Moreover, CNN's are very bad at encoding different representations of pose and orientation within themselves. In CNNs, the orientations and their surroundings are not taken into account.

7.5.1 CNN Algorithm

Algorithm for CNN-based Classification

1. Apply the convolution filter in the first layer
2. The sensitivity of the filter is reduced by smoothing
   the convolution filter (i.e) subsampling
3. The signal transfers from one layer to another
   layer is controlled by the activation layer
4. Fasten the training period by using rectified
   linear unit (RELU)
5. The neurons in the proceeding layer are connected to
   every neuron in the subsequent layer
6. During training Loss layer is added at the end to
   give feedback to the neural network

Algorithm 1: Accuracy of the proposed CNN model

1. loadImage();

2. dataAugmentation();

3. splitData();

4. laodData();

5. for each epoch in epochNumber do

6. for each batch in batchSize do

7. ˆy = model(features);

8. loss = crossEntropy(y, ˆy);

9. optimization(loss);

10. Accuracy = (1 - loss) * 100%;

11. end

12. end

8. CODE

CNN code:

```
from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense

from tensorflow.keras.layers import Convolution2D

from tensorflow.keras.layers import MaxPooling2D

from tensorflow.keras.layers import Flatten

from keras.preprocessing.image import ImageDataGenerator


train_datagen =  ImageDataGenerator(rescale = 1./255,shear_range = 0.2,zoom_range = 0.2,horizontal_flip = True)

test_datagen =  ImageDataGenerator(rescale = 1./255)


x_train = train_datagen.flow_from_directory(r “C:\Users\piyus\OneDrive\Desktop\Mini Project\Brain-Tumor-Prediction-using-CNN-main\”BTD,target_size = (64,64),batch_size = 32, class_mode ="categorical")

x_test = test_datagen.flow_from_directory(r" C:\Users\piyus\OneDrive\Desktop\Mini Project\Brain-Tumor-Prediction-using-CNN-main\BTD",target_size = (64,64),batch_size = 32, class_mode ="categorical")


model = Sequential()

model.add(Convolution2D(32,(3,3), input_shape=(64,64,3)))

model.add(MaxPooling2D(2,2))

model.add(Flatten())


model.add(Dense(units = 256,activation = "relu",kernel_initializer = "random_uniform"))

model.add(Dense(units = 256,activation = "relu",kernel_initializer = "random_uniform"))

model.add(Dense(units = 2,activation = "softmax",kernel_initializer = "random_uniform"))

model.compile("sgd",loss = "categorical_crossentropy",metrics = ["accuracy"])
```

```
model.fit_generator(x_train ,steps_per_epoch = 7 ,epochs = 100,validation_data= x_test ,
validation_steps = 1)

model.save("BTD2.h5")
```

Prediction code

```
from tensorflow.keras.models import load_model

from tensorflow.keras.preprocessing import image

import cv2

import numpy as np

index = ["no","yes"]

model = load_model(r"C:\Users\piyus\OneDrive\Desktop\Mini Project\Brain-Tumor-
Prediction-using-CNN-main\BTD2.h5")

img = image.load_img(r" C:\Users\piyus\OneDrive\Desktop\Mini Project\Brain-Tumor-
Prediction-using-CNN-main\Testing images",target_size = (64,64))

type(img)

x = image.img_to_array(img)

x = np.expand_dims(x,axis=0)

pred = model.predict_classes(x)

pred

print(index[pred[0]])
```

Flask code

```
@app.route('/',methods=["GET","POST"])
def display():
    form = MRIFileForm()
    if request.method == "GET":
        return render_template("index.html",form=form)
    elif request.method == "POST":
        if form.validate_on_submit():
            file = form.file.data
            saved_picture = save_picture(file)
            result = prediction(saved_picture)
            if result == "no":
                alert = "success"
                flash('No tumor has been Detected!!',alert)
            elif result == "yes":
                alert="danger"
                flash('Tumor has been Detected!!',alert)
            return render_template('index.html',form=form)



      if __name__ == '__main__':
      app.run(debug = True)
```

9. Outputs :

```
Epoch 1/120

C:\Users\piyus\AppData\Local\Temp\ipykernel_35528\4200880667.py:1: UserWarning: `Model.fit_generator` is deprecated and
will be removed in a future version. Please use `Model.fit`, which supports generators.
  model.fit_generator(x_train ,steps_per_epoch = 7 ,epochs = 120,validation_data= x_test , validation_steps = 1)

Output exceeds the size limit. Open the full output data in a text editor
7/7 [==============================] - 1s 121ms/step - loss: 0.3407 - accuracy: 0.8491 - val_loss: 0.4700 - val_accuracy:
0.8125
Epoch 2/120
7/7 [==============================] - 1s 126ms/step - loss: 0.2988 - accuracy: 0.8868 - val_loss: 0.6255 - val_accuracy:
0.8125
Epoch 3/120
7/7 [==============================] - 1s 142ms/step - loss: 0.2952 - accuracy: 0.8632 - val_loss: 0.4904 - val_accuracy:
0.8125
Epoch 4/120
7/7 [==============================] - 1s 119ms/step - loss: 0.2964 - accuracy: 0.8868 - val_loss: 0.5183 - val_accuracy:
0.8438
Epoch 5/120
7/7 [==============================] - 1s 113ms/step - loss: 0.2848 - accuracy: 0.8726 - val_loss: 0.3763 - val_accuracy:
0.8750
Epoch 6/120
7/7 [==============================] - 1s 113ms/step - loss: 0.3068 - accuracy: 0.8821 - val_loss: 0.5669 - val_accuracy:
0.8125
```

Figure 8: Output 1

```
Epoch 9/120
7/7 [==============================] - 1s 121ms/step - loss: 0.2680 - accuracy: 0.8915 - val_loss: 0.5098 - val_accuracy:
0.8125
Epoch 10/120
7/7 [==============================] - 1s 115ms/step - loss: 0.3030 - accuracy: 0.8774 - val_loss: 0.4267 - val_accuracy:
0.7812
Epoch 11/120
7/7 [==============================] - 1s 112ms/step - loss: 0.3065 - accuracy: 0.8962 - val_loss: 0.5779 - val_accuracy:
0.7500
Epoch 12/120
7/7 [==============================] - 1s 117ms/step - loss: 0.3087 - accuracy: 0.8868 - val_loss: 0.3887 - val_accuracy:
0.8750
Epoch 13/120
7/7 [==============================] - 1s 111ms/step - loss: 0.3141 - accuracy: 0.8443 - val_loss: 0.3822 - val_accuracy:
0.8750
...
Epoch 119/120
7/7 [==============================] - 1s 146ms/step - loss: 0.1314 - accuracy: 0.9481 - val_loss: 0.4736 - val_accuracy:
0.8125
Epoch 120/120
7/7 [==============================] - 1s 119ms/step - loss: 0.1344 - accuracy: 0.9481 - val_loss: 0.9649 - val_accuracy:
0.8438

<keras.callbacks.History at 0x224debe2620>
```

Figure 9: Output 1

Frontend output :

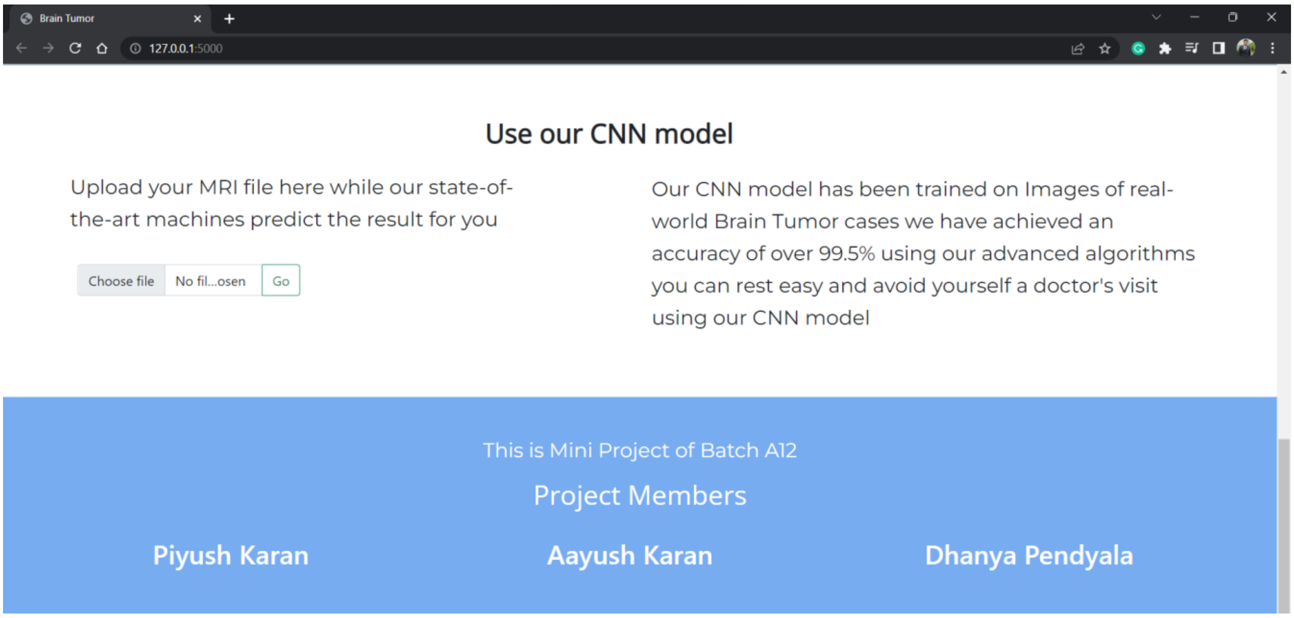


Figure 10: GUI of the site(Frontend output)

10. MODULES

Module 1 Description: Preprocessing the data

The first step is to collect the data. Then we segregate the acquired data into training data and testing data. We're going to use the 10-fold clustering method to segregate the same using K-Means.

After that, The training data is allocated at 80%. The test data is the remaining 20%.

Module 2: Training the model using CNN

In this module we train our model which is built using the CNN algorithm given the data which we have pre-processed.

Firstly we import the pre-processed data into our CNN algorithm i.e to the Convolutional Layer – In which the process of extracting valuable features from an image is done and then the convolution layer has several filters that perform the convolution operation. Every image is considered a matrix of pixel values.

The output of the Convolutional Layer will be the input of our second layer i.e Pooling Layer- here Pooling is a down-sampling operation that reduces the dimensionality of the feature map. The rectified feature map goes through a pooling layer to generate a feature map. We have used the ReLU activation function.

Here the output of the pooling layer will be the input of our third layer i.e Fully Connected layer – In this layer, the input from the other layers will be flattened into a vector and sent. It will transform the output into the desired number of classes by the network.

Once the model is trained with the data we do the epoch multiple times so that our model learns every time we train it through a concept called transfer learning.

Module 3: Front-end design

As part of module 3, we're going to create a webpage to display the output by taking input from the user.

On our web page, the users are provided with an option to upload the MIR report as input.

Through our model processes it and displays the result i.e the output will be shown if the brain tumor is detected or not in a flash.

11. Conclusion and Future Works

11.1 Conclusion

Without the pre-trained Keras model, the training accuracy is 100 % and the validation accuracy is >90%. The validation result had the best figure of 100% accuracy. It is observed that without using the pre-trained Keras model, although the training accuracy is >90%, the overall accuracy is low unlike where the pre-trained model is used.

Also, when we trained our dataset without Transfer learning, the computation time was 40 min whereas when we used Transfer Learning, the computation time was 15min. Hence, training and computation time with the pre-trained Keras model was 50% lesser than without. The chances of overfitting the dataset is higher when training the model from scratch rather than using pre-trained Keras. Keras also provides an easy interface for data augmentation. Amongst the Keras models, it is seen that ResNet 50 has the best overall accuracy as well as F1 score. ResNet is a powerful backbone model that is used very frequently in many computer vision tasks. Precision and Recall both cannot be improved as one comes at the cost of the other. So, we use the F1 score too.

11.2 Future Work

Build an app-based user interface in hospitals that allow doctors to easily determine the impact of tumor and suggest treatment accordingly Since the performance and complexity of ConvNets depend on the input data representation we can try to predict the location as well as stage of the tumor from Volume based 3D images. By creating three-dimensional (3D) anatomical models from individual patients, training, planning, and computer guidance during surgery are improved. Using VolumeNet with LOPO (Leave-One-Patient-Out) scheme has proved to give a high training as well as validation accuracy(>95%).In the LOPO test scheme, in each iteration, one patient is used for testing and the remaining patients are used for training the ConvNets, this iterates for each patient.

Although the LOPO test scheme is computationally expensive, using this we can have more training data which is required for ConvNets training.

LOPO testing is robust and most applicable to our application, where we get test results for each individual patient. So, if the classifier misclassifies a patient then we can further investigate it separately.

Improve testing accuracy and computation time by using classifier boosting techniques like using more number images with more data augmentation, fine-tuning hyperparameters, training for a longer time i.e. using more epochs, adding more appropriate layers, etc.. Classifier boosting is done by building a model from the training data and then creating a second model that attempts to correct the errors from the first model for a faster prognosis. Such techniques can be used to raise the accuracy even higher and reach a level that will allow this tool to be a significant asset to any medical facility dealing with brain tumors.

For more complex datasets, we can use U-Net architecture rather than CNN where the max-pooling layers are just replaced by up-sampling ones. Ultimately we would like to use very large and deep convolutional nets on video sequences where the temporal structure provides very helpful information that is missing or far less obvious in static images.

Unsupervised transfer learning may attract more and more attention in the future.

12. REFERENCES

1) S. Bauer et al., —A survey of MRI-based medical image analysis for brain tumor studies,‖ Phys. Med. Biol., vol. 58, no. 13, pp.97–129, 2013.
2) B. Menze et al., —The multimodal brain tumor image segmentation benchmark (BRATS),‖ IEEE Trans. Med. Imag., vol. 34, no.10, pp. 1993–2024, Oct. 2015
3) B. H. Menze et al., —A generative model for brain tumor segmentation in multi-modal images,‖ in Medical Image Computing and Comput.- Assisted Intervention-MICCAI 2010. New York: Springer, 2010, pp. 151–159
4) S. Bauer, L.-P. Nolte, and M. Reyes, —Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization,‖ in Medical Image Computing and Comput.-Assisted Intervention-MICCAI 2011. New York: Springer, 2011, pp. 354–361.
5) C.-H. Lee et al., —Segmenting brain tumors using pseudo-conditional random fields,‖ in Medical Image Computing and Comput.-Assisted
6) Intervention-MICCAI 2008. New York: Springer, 2008, pp. 359–366
7) R. Meier et al., —A hybrid model for multimodal brain tumor segmentation,‖ in Proc. NCIMICCAI BRATS, 2013, pp. 31–37.
8) Vinod Kumar, JainySachdeva, Indra Gupta —Classification of brain tumors using PCAANN‖ 2011 World Congress on Information and Communication Technologies.
9) Sergio Pereira, Adriano Pinto, Victor Alves, and Carlos A. Silva —Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images‖IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 35, NO. 5, MAY 2016
10) RaselAhmmed, Md. Foisal Hossain —Tumor Detection in Brain MRI Image Using Template based K-means and Fuzzy C-means Clustering Algorithm‖ 2016 International Conference on Computer Communication and Informatics (ICCCI -2016), Jan. 07 –09, 2016, Coimbatore, INDIA
11) S.C. Turaga, J.F. Murray, V. Jain, F. Roth, M. Helmstaedter, K. Briggman,W. Denk, and H.S. Seung. Convolutional networks can learn to generate affinity graphs for image segmentation. Neural Computation, 22(2):511–538, 2010