Early Detection of Brain Tumor using Capsule Network

BY

MEHEDI HASAN
ID: 163-15-1118

MD. ABUL HAYAT
ID: 163-15-1132
AND
SHAYLA ALAM SETU
ID: 163-15-1150

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

MD REDUANUL HAQUE

Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

MOHAMMAD JAHANGIR ALAM

Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

OCTOBER 2020

APPROVAL

This Project/internship titled **"EARLY DETECTION OF BRAIN TUMOR USING CAPSULE NETWORK"**, submitted by Mehedi Hasan , ID No:163-15-1118, Md. Abul Hayat, ID No:163-15-1132, Shayla Alam Setu, ID No:163-15-1150,to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **24-09-2020**.

BOARD OF EXAMINERS

____

Dr. Syed Akhter Hossain **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. S M Aminul Haque **Internal Examiner**
Associate Professor & Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Md. Ohidujjaman **Internal Examiner**
Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

____

Dr. Mohammad Shorif Uddin **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **MD REDUANUL HAQUE, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

MD REDUANUL HAQUE
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

MOHAMMAD JAHANGIR ALAM
Lecturer
Department of CSE
Daffodil International University

Submitted by:

Mehedi

MEHEDI HASAN
ID: -163-15-1118
Department of CSE
Daffodil International Universitv

Hayat

MD. ABUL HAYAT
ID: -163-15-1132
Department of CSE
Daffodil International University

Setu

SHAYLA ALAM SETU
ID: -163-15-1150
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound indebtedness to **Md Reduanul Haque**, **Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of **"**Deep Learning**"** to carry out this project. His endless patience, scholarly guidance, continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. S M Aminul Haque**, **Associate Professor and Associate Head,** Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

The brain tumor is one of the deadliest diseases in the world nowadays. Only in the United States of America, today the number of people having brain tumor is more than 700,000 [1]. Approximately 16,000 people would die in the process of a brain tumor in the year 2020 [1]. It'll be really grateful for monitoring and identification if the characterization of tumors in the brain can be done at a very pre-mature stage. Numerous researchers have already taken some attempts to use various techniques, such as digital mammography, MRI, CT (Computed Tomography), etc. To detect the exact type of brain tumor from MRI images CapsNets became an improved architecture. Since these networks can operate with fewer training samples. We used a dataset from kaggle to monitor the tumor in the brain at the very initial stage. AT first, in the CNN model, each of the input pictures will move through a set of filter convolution layers (called Kernels), then pooling, then completely related layers (FC) and applying Soft-max function to define a probabilistic meaning object. The outcome from the proposed technique reveals that 92 percent of accuracy can be gained from this technique.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

1.1 Overview

A tumor which begins in the spinal cord or brain is a primary spinal cord or brain tumor. An estimated 23,890 adults (13,590 men and 10,300 women) will be diagnosed with primary cancer of the brain and spinal cord in the United States this year 2020 [2]. The probability of a person developing this kind of tumor in their lifetime is less than 1%. Of all main central nervous system (CNS) tumors, brain tumors account for 85 percent to 90 percent [3]. This year, about 3,540 kids under the age of 15 will also be diagnosed with a brain or CNS tumor. The remainder of this guide deals with primary adult brain tumors [3].

The most prevalent and violent disease is brain tumors, leading to a very low life span at the highest level. Thus, recovery preparation is a crucial step in improving patients ' quality of life. Different types of imaging methods like computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound imaging are commonly used to test tumors in the brain, lung, liver, breast, prostate, etc. In particular, MRI images are used in this work to diagnose tumors in the brain. The enormous amount of data produced by the MRI scan, however, thwarts manual tumor vs. non-tumor classification at a specific time. However, for a limited number of images, precise quantitative measurements are given with some limitations. In order to avoid human death rates, a trustworthy and automatic classification scheme is therefore necessary. In the broad spatial and structural heterogeneity of the surrounding brain tumor area, the automatic classification of a brain tumor is a very difficult task. Automatic brain tumor detection is proposed in this study using the classification of Convolutional Neural Networks (CNN). By using small kernels, the deeper architecture design is carried out. The neuron weight is given as a small number. Experimental findings show that the 97.5 percent accuracy of the CNN archives is poor in difficulty relative to all other states of the art methods [4].

CNNs are active nine times out of ten, as we hear regarding deep learning breaching a new technical barrier. In certain examples, they have learned to organize photos into classifications much better than humans. If there is one approach that justifies the hype out there, it is CNNs. What's really cool about them is that, at least when you break them down into their basic pieces, they are easy to comprehend.

Images are compared piece by piece by CNNs. The parts that it scans are called characteristics. CNNs get much better at seeing similarity than whole-image matching schemes by finding rough function matches in essentially the same places in two pictures.

1.2 Problem Definition

Brain tumor identification is a major problem in medical science. If we can identify brain tumors at the early stage using machine learning, we can reduce the mortality rate drastically. Sometimes convolutional neural networks may give less accurate results because of data shortage. For the most part capsule network solve this problem.

1.3 Motivation

There has recently been a surge of interest in Capsule Networks (CapsNets) for the issue of the classification of brain tumor type. To detect the exact type of brain tumor from MRI images CapsNets became an improved architecture of CNN that gives more accurate results than other machine learning methods. It’s our duty to develop a more accurate and efficient technique to handle brain tumor type classification. By seeing people’s suffering, we get more motivation of using this CapsNets architecture to detect brain tumor type classification. The report will cover data acquisition, image processing, and future works.

1.4 Objectives

To create an improved architecture that maximizes the accuracy of the classification issues at hand for the brain tumor classification problem, implement and integrate Capsule Networks (CapsNets). Investigate the over-fitting problem of CapsNets based on a true set of MRI images. Start exploring whether or not CapsNets will provide a better fit for all brain images or just the specific segmented tumor. Develop a visualization paradigm of CapsNet output to help explain the characteristics taught. So we're presenting that an evaluation of CNN models including LeNet-5, VGG 16, Inception V1, ResNet, Xception, and AlexNet to achieve a better recognition accuracy compared too many exiting approaches.

CHAPTER 2
Literature Review

2.1 Literature Review

Brain tumor identification is an indispensable way to treat brain tumor disease. Convolutional Neural Network is a widely used method in this field. In our work, we have used one of CNN architecture named Capsule Networks (CapsNets). Among different types of deep CNN methods proposed by many researchers, CapsNets provide the most efficient result. CapsNets can work on a small dataset and its accuracy rate is high, that's why we are working on this CNN architecture. We also studied some renowned scholar's works in this field and their achievements.

Havaei et al. [5]'s recent research for brain tumor segmentation. It proposed a deep Convolutional Neural Network (CNN) two-path that takes into account the properties of the pixels and surveys the neighboring pixel's probabilities. When the region of the tumor segmentation is completed, it is possible to extract various kinds of features to be served to the identification module.

Usman et al. [6] used these input vector features such as intensity, differences in intensity wavelet texture, and neighborhood to instruct a random forest classifier. These effects of tumor area augmentation are Strength histogram, Bag-of-Words (BoW), and Gray Degree Co-occurrence Matrix (GLCM) [7]. The specificity of brain tumor classification can be improved by tumor region augmentation.

Abbadi et al. [8] have hypothesized that the Gray Level Run Length Matrices (GLRLM) and GLCM can be used to eliminate eighteen tumor classification features by Probabilistic Neural Networks (PNN) for these two items. All previously described tumor classification studies have had one significant downside, i.e. they require advanced knowledge of the type of functionality to be eliminated, which also lowers their capacity for generalization.

There is tremendous learning potential for CNNs [9]. Without any initial information, it can also collect the essence of an input image, providing a satisfactory image classification system for them. Recently, the deployment of CNNs for the identification of the type of brain tumor is discussed in [10], where both CNNs and neural networks are used along with various pre-processing measures that involve augmentation of data. CNNs which are functional without any pre-processing surpass other MRI techniques for the axial brain. CNNs have certainly overcome several image processing methods, but they also have some disadvantages. For example, they are not powerful enough to influence transformation yet do not take into account the spatial relations within the picture. Therefore, in order to strengthen their generalization, CNNs must be provided for training data that includes all sorts of rotations. In addition, small data sets are typically poorly opposed by CNNs, most of which are medical image collections, like brain MRIs.

Capsule networks (CapsNets) have recently been proposed by Sabour and Hinton et al. [11] to address the previously stated disadvantages of CNNs. There are multiple neurons for every capsule inside the network. Numerous position parameters such as position, scaling, orientation regulate the activity vector of every capsule. The probability of the particular object's presence described by the capsule provides the duration of each vector of operation. Routing using agreement is the most important aspect of CapsNets, which means that capsules at lower levels forecast the outcome of capsules at higher levels, and the higher-level capsules only start if these predictions agree. Then we optimize the attainability of these advantages in this study and support the CapsNet structure for the issue of detecting brain tumor type. Throughout this regard, we look at many possible Capsule network architectures and consider the one that improves the precision of the prediction for the issue at hand. In addition, to investigate the impact of input data on Caps Networks, we analyze the following two situations as the input to the built CapsNets (i) entire image of the brain is presented to that network, as well as; (ii) segmented areas are absorbed. Because of the complex structures of CapsNets, it is important to learn many parameters, they appear to train data over-fitting mainly on tiny datasets such as brain MRIs. As a result, the overfitting issue of Caps Nets for the brain tumor identification problem is considered to resolve a regularization criterion. At last, we extend the visualization framework for the outcome of the Caps Nets to properly evaluate features that the built model developed using the data.

The remaining of the whole paper is designed like this: the mathematical context needed for CNNs is described in section 3. Section 3 takes Caps Nets into account and introduces in Section 4 the suggested solution followed by experimental outcomes. Finally, the paper ends with Section 5.

“By using Convolution Neural Networks (CNN) in the work, automatic brain tumor detection is suggested by J Seetha et al” [4]. The deeper architecture implementation is carried out by using tiny kernels. “They used neurons in tiny sizes. We can see from experimental findings that the CNN registers 97.5 % which has very low complexity and also contrasts it with the remaining states of the art method ” [4].

The idea of deep transfer learning is taken up in the [12] proposed classification system and uses a GoogLeNet that is pre-trained so that features can be extracted from MRI images of the brain. To classify the removed functions, the classifier models which are already proven were unified. “After a five-fold cross-validation procedure at the patient level, the experiment records a mean classification on the fishare. Suggested framework MRI dataset that gives 98 percent precision, surpassing all state-of-the-art methods” [12].

CHAPTER 3

Methodology

3.1 Convolutional Neural Network

The CNN (Convolutional Neural Network) is a deep learning algorithm for image processing. For most of the machine learning practitioners today, CNN is an essential instrument. An input image is taken from CNN image classifications, processed, and categorized under those categories. Technically, the preparation and testing of CNN deep learning models, each of the input pictures will move through a set of filter convolution layers (called Kernels), then pooling, then completely related layers (FC) and applying Soft-max function to define a probabilistic meaning object between 0 and 1.

The following three properties are used by CNNs: First of all, every single layer gets feedback from the previous phases. Using this method, it is possible to detach the edges and corners. After that, the characteristics may be combined to classify higher-order characteristics in the next layers. Then the principle of shared weights is the second important parameter, which means that for the overall picture, similar feature detectors can be used. Ultimately, CNNs typically got many levels of sub-sampling. These layers are centered on the issue that not only beneficial but also harmful are the exact position of the characteristics, as for various causes, this detail tends to vary. [13].

The following figure 3.1 demonstrates a full CNN flow for processing an input image and classifying the objects according to values.

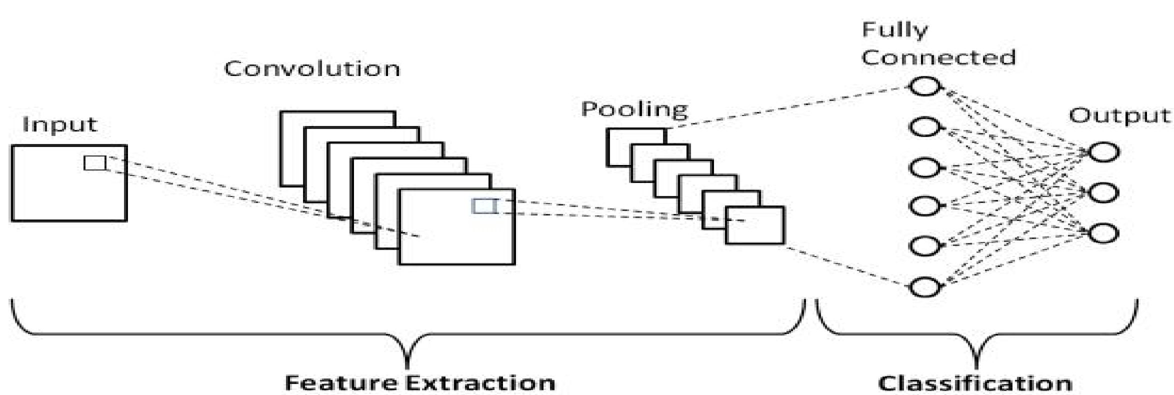


Fig 3.1: Schematic diagram of a basic convolutional neural network

Although CNN is very useful in many areas, it has some fundamental drawbacks. CNN does not encode the object's location and orientation. The ability to be spatially invariant to the input data is missing.

Let us consider an instance that is very simple and non-technical. Imagine the face. What components are there? We have an oval profile, two eyes, a mouth, and a nose. The mere existence of these objects can be a very strong indication for a CNN to take into account that there is a face in the picture. Orientation and relative spatial relationships are not very important to a C between these components

"The pooling operation that used in convolutionary neural networks being a major mistake and the fact that it works that much well is a misfortune," Hinton said.

To overcome these drawbacks a most recent architecture called Capsule Network (CapsNet) is introduced.

3.2 Different Types of CNN

3.2.1 LeNet-5

Among the most commonly recognized CNN architecture is the LeNet-5 architecture. It was founded by Yann LeCun in 1998. Two groups of convolutional and averaged pooling layers consist of the LeNet-5 architecture, preceded by a flattening convolutional layer, then two completely linked layers, and finally a softmax classifier.

The following figure 3.2.1 is shows LeNet-5 Architecture

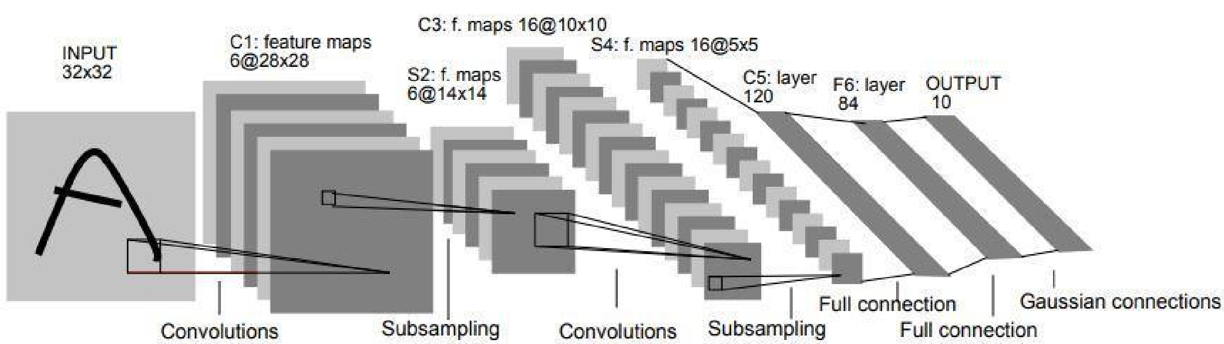


Figure 3.2.1: Architecture of LeNet -5

The input is an image of size 32×32. The original MNIST images are 28×28 in size, but for the input layer, they are zero-padded to 32×32. The explanation for this is increasing the size of the input image lets the network acknowledge the stroke endpoints and corners of the images better.

Then the first convolutional layer of size 28×28. The kernel size is 5×5 and it has a tanh activation function. After that, the average layer called pooling is present with a kernel with a size of 2×2 and the same tanh activation function. This layer actually reduces the size of the previous convolutional layer into half. The layer before it was 28×28, but then, the pooling layer reduces it to 14×14. The same principle follows for two more layers. Before the two fully connected layers we have a convolutional layer that has 120 featured maps and having the size 1×1. Here, the kernel size and activation functions are 5×5 and tanh respectively.

The last two layers are fully connected dense layers. The first completely linked layer has 84 units and also has tanh activation function. Now, the last layer has 10 units which correspond to each of the 10 digits (from 0 to 9) for the MNIST data set. Here, the activation function softmax is very common in many of the output layers in other networks as well, especially for classification purposes.

Table I: Brief Architecture of LeNet-5

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	32x32	-	-	-
1	Convolution	6	28x28	5x5	1	tanh
2	Average Pooling	6	14x14	2x2	2	tanh
3	Convolution	16	10x10	5x5	1	tanh
4	Average Pooling	16	5x5	2x2	2	tanh
5	Convolution	120	1x1	5x5	1	tanh
6	FC	-	84	-	-	tanh
Output	FC	-	10	-	-	softmax

3.2.2 AlexNet

Five convolutional layers consist of the Alex-Net architecture, several of Maximum pooling layers are followed and then two completely linked layers and finally a 1000 way softmax classifier.

The following figure 3.2.2 demonstrates Architecture of Alex-Net

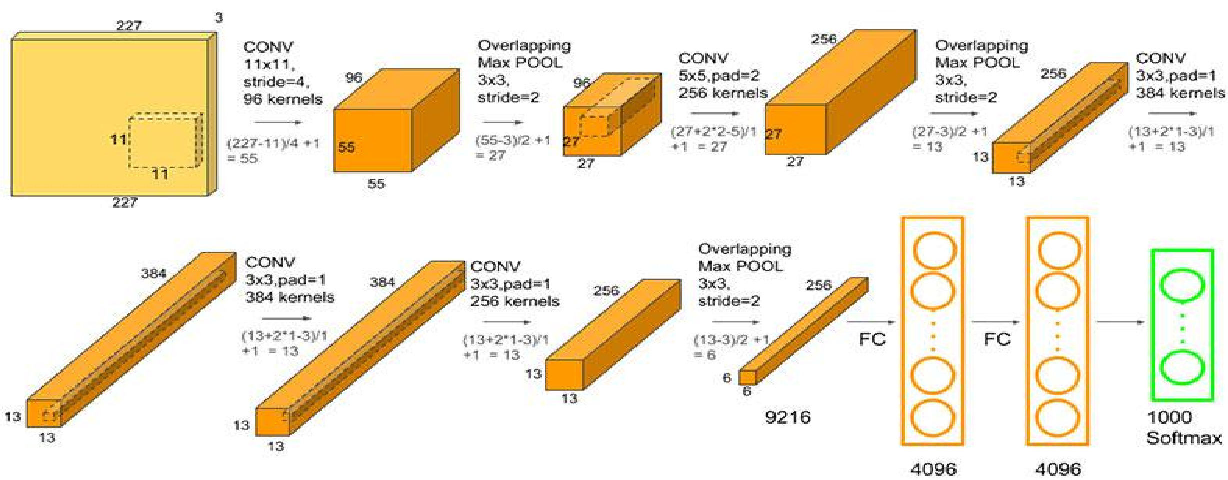


Fig 3.2.2: Architecture of Alex-Net

The Alex-Net input is a 227x227x3 RGB image that the first passes through the convolution Layer with 96 feature maps or filters with a size of 11 to 11 and a 4th phase. The dimension of the image will change to 55x55x96.

The maximum layer for pooling or the sub-sampling layer with a filter size of 3 to 3 and a step of two is then added to the Alex-Net. The image dimensions resulting from this will be minimized to 27x27x96.

Next, there's a second one convolution layer with 256 characteristic maps with a scale of 5 to 5 and a step of 1. Then, with a filter size of 3x3, the full pooling layer and a stage of 2 is again available.

Except that it has 256 function maps, this layer is similar to the second layer. So the performance will be decreased to 13x13x256.

Convolutional filter-sized layers 3x3 and a phase of 1 are the third, fourth, and fifth layers. 384 feature maps were used for the first two, with 256 filters used for the third.

Max layer of pooling with filter size 3-3, a phase of 2, and 256 characteristic maps follow the three convolutional layers.

The convolutionary layer output in the sixth layer is flattened by a fully associated layer with feature maps of 9216 of 1 to 1 size each. Two fully-connected 4096 unit layers are again in the next seven and eight layers. Finally, with 1000 possible values, there is a softmax performance layer.

Table II: Brief Architecture of Alex-Net

Layer		Feature Map	Size	Kernel Size	Stride	Activation
Input	Image	1	227x227x3	-	-	-
1	Convolution	96	55 x 55 x 96	11x11	4	relu
	Max Pooling	96	27 x 27 x 96	3x3	2	relu
2	Convolution	256	27 x 27 x 256	5x5	1	relu
	Max Pooling	256	13 x 13 x 256	3x3	2	relu
3	Convolution	384	13 x 13 x 384	3x3	1	relu
4	Convolution	384	13 x 13 x 384	3x3	1	relu
5	Convolution	256	13 x 13 x 256	3x3	1	relu
	Max Pooling	256	6 x 6 x 256	3x3	2	relu
6	FC	-	9216	-	-	relu
7	FC	-	4096	-	-	relu
8	FC	-	4096	-	-	relu
Output	FC	-	1000	-	-	Softmax

3.2.3 VGG16

VGG16 consists of 16 convolutional layers**.** The Relu activation feature of Alex-Net has thirteen convolutional layers and three fully connected layers.

The following figure 3.2.3 demonstrates Diagram of VGG16

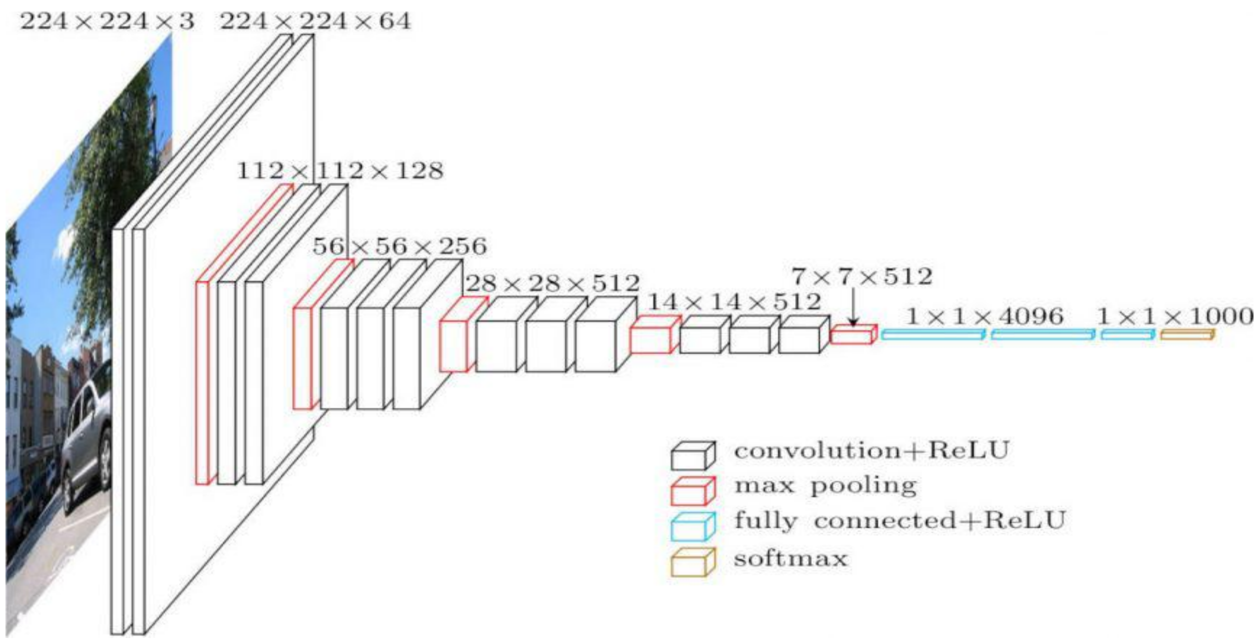


Figure 3.2.3: Diagram of VGG16

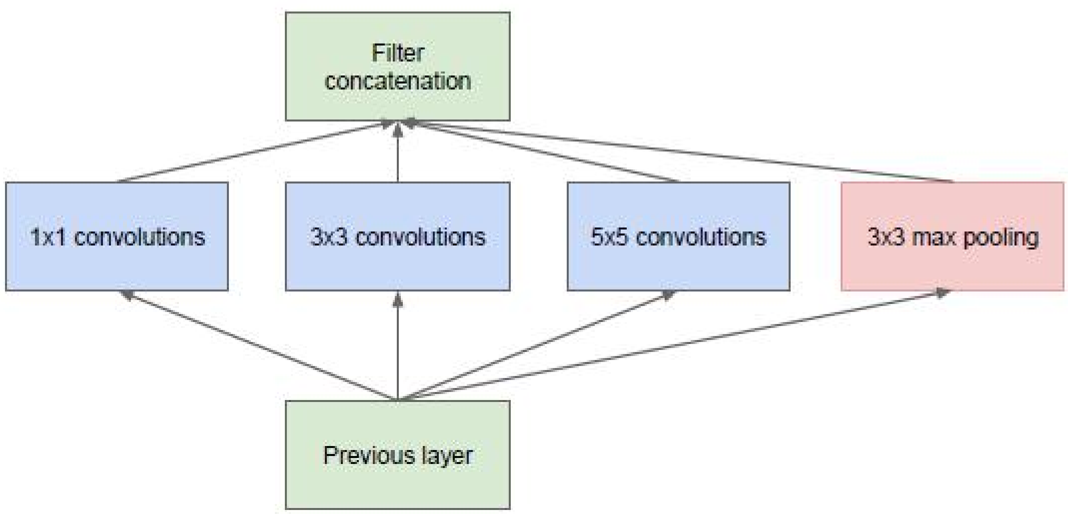
The input produces to cov1 layer is size 224 x 224 RGB images which are fixed. A convolutional (conv.) layers stack is been passed through the image where a filter has been used. The filters along with a limited Responsive field: 3 to 3 (which is the lowest size to be caught the up / down, left / right, center concept). It uses a 1x1 convolution filters in a setting, that can be shown as a linear transformation (followed by non-linearity) of the input channels. The convolution stage is set to 1 pixel; spatial padding being inputted by the convolutional layer is such that after convolution the spatial resolution is retained, that is a for 3-3 convolution layers the padding is 1-pixel. Five max-pooling layers, which obey some of that convolution layers, conduct spatial pooling. With step 2, over such a 2 to 2-pixel window, max-pooling is carried out.

Three Fully-Connected (FC) layers follow a stack of such convolutional layers featuring different depths throughout different architectures. Among them the earliest 2 have 8192 channels, the 3rd carries out 1000 way ILSVRC classification including 1000 channels (1 for every class). The soft-max layer is the final layer. On all networks, the specification of the layers which are fully connected remains the same.

3.2.4 Inception-V1

Basically, Inception v1 is a convolutional neural network (CNN) that is 27 layers wide. 1-1 Convolutionary layer before another layer is added, which is primarily used for dimensionality reduction.

The following figure 3.2.4(a) demonstrates the inception module (naive version, without 1×1 convolution)



(a) Inception module, naïve version

Figure 3.2.4(a): Inception Module

For each layer, Conv size is previously set, such as Alex-Net, and VGG-Net. Now, for the previous input, 1-1 Conv, 3-3 Conv, 5-5 Conv, and 3-3 max-pooling are performed entirely, stacking with each other again at the output. Various convolution sizes as well as max-pooling are attempted when the picture comes in. Then various types of features are extracted.

After that, all function maps, the input is concatenated as the input of the next module on different paths.

Without the 1-1 convolution as above, however, we can imagine how great the number of operations is!

The following figure 3.2.4(b) demonstrates the Inception module with reduction in dimensionality

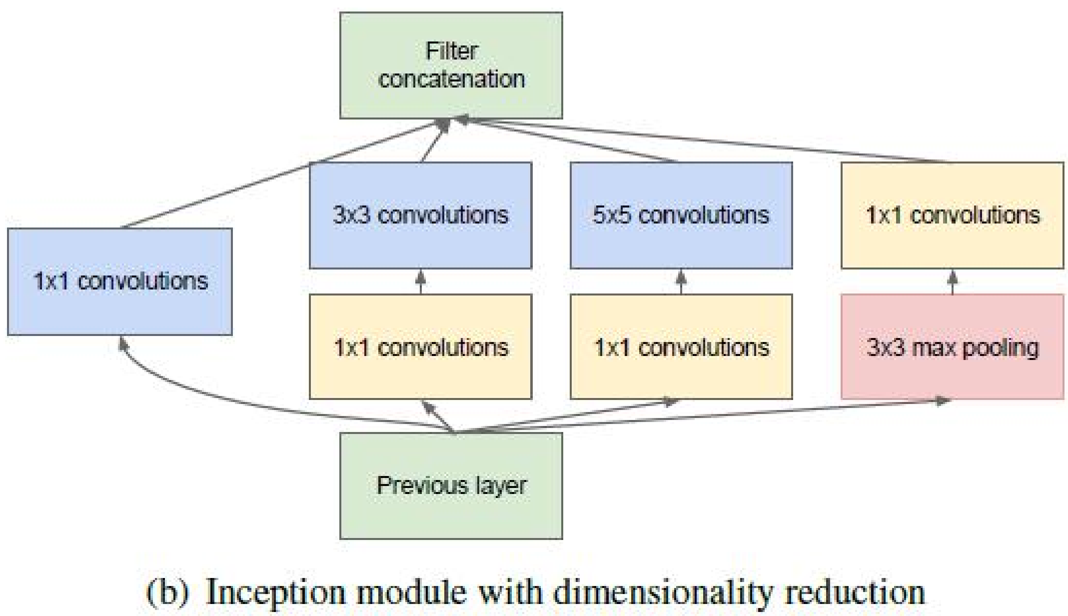


Figure 3.2.4(b): Inception module with reduction in dimensionality

Thus, for dimension reduction, 1 to 1 convolution is introduced into the inception module!

3.2.5 ResNet

The Residual Network (Res-Net) is Convolutional Neural Network's (CNN) one of architecture that generates hundreds of convolutionary layers. We had the effectiveness of additional layers in previous CNN architectures, but in Res-Net, it can add a large number of layers with good results.

For the "vanishing gradient" problem, Res-Net was an innovative solution. Neural networks train via the back-propagation process that looks for the weights that reduce it, gradient descent is relied upon by it, and the loss function is heading down. For so many layers gradient is getting smaller continuously by repeated multiplication. Until the gradient totally disappears, iteration keeps running. As a result, output saturates with every additional layer.

"Identity shortcut links" is the remedy for Res-Net. Res-Net stacks mappings for identity, Layers which initially do nothing, and that skip past them, then reuse prior layer activations. Initially, the network is compressed through skipping into just several layers, allowing for quicker learning. As the network operates again, these layers are expanded, exploring the "residual" parts of that same network with more of the feature space of the source picture.

The following figure 3.2.5 demonstrates Res-Net Architecture

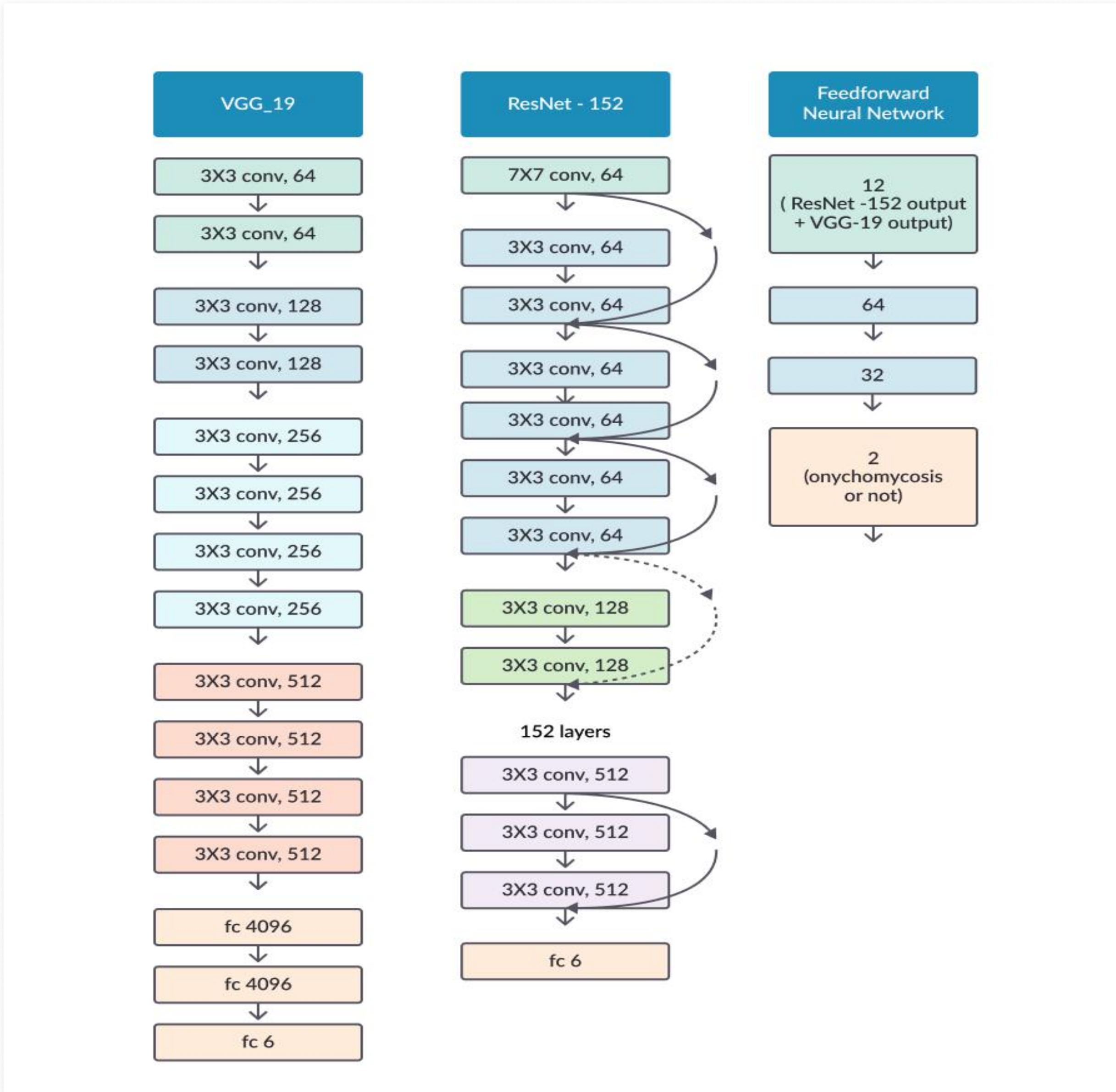


Figure 3.2.5: Res-Net Architecture

3.2.6 Xception

Xception is a profound convolutionary architecture of the neural network which requires Separable Depth Wise Convolutions and focuses on three sections: Entry, Middle, and Exit Flow. Researchers at Google created it. In convolutional neural networks, Google proposed an explanation of Inception modules as an intermediate stage between normal convolution and the convolution activity separable by depth.

The following figure 3.2.6 Shows architecture of Xception

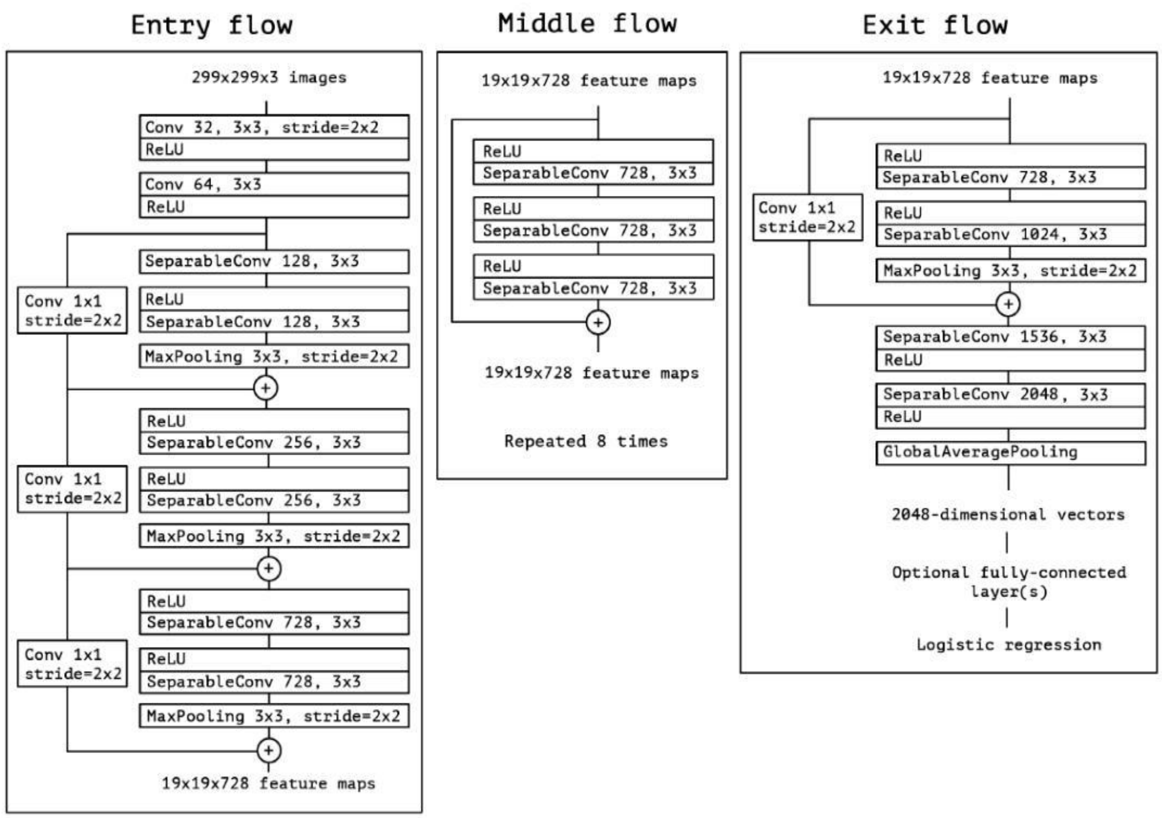


Figure 3.2.6: Architecture of Xception

The data first goes through the flow of entry, then through the middle flow that is performed eight times, then finally into the flow of exit. Both separable Convolution and convolutional layers are accompanied by batch normalization, to be noted. In most classical classification problems, the Xception architecture have passed Inception V3, VGG-16 and Res-Net.

3.2.7 ResNeXt

ResNeXt is a homogeneous neural network that reduces the number of hyper parameters which requires conventional ResNet. This is achieved by their use of "cardinality". Cardinality specifies the size of the set of transformations.

The following figure 3.2.7 Shows ResNeXt's block diagram

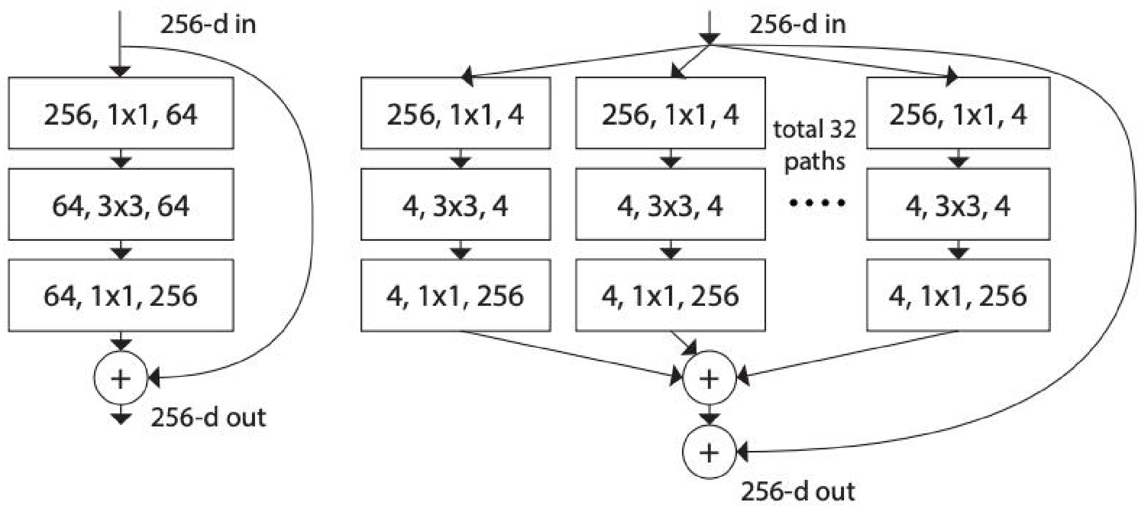


Figure 3.2.7: ResNeXt's block diagram

In this image the leftmost diagram is a conventional Res-Net block; the rightmost is the ResNeXt block, which has a cardinality of 32. The same transformations are applied 32 times, and the result is aggregated at the end.

There are two laws outlining the fundamental architecture of ResNeXt. Second, if the blocks generate spatial maps of the same dimensions, they provide the same collection of hyper parameters, so if the spatial maps are filtered down by a factor of two, a factor of two multiplied the block width.

Table III: Brief block structure of ResNeXt

stage	output	ResNet-50	**ResNeXt-50 (32×4d)**
conv1	112×112	7×7, 64, stride 2	7×7, 64, stride 2
conv2	56×56	3×3 max pool, stride 2	3×3 max pool, stride 2
		$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128, C\text{=}32 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3	28×28	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256, C\text{=}32 \\ 1\times1, 512 \end{bmatrix}\times4$
conv4	14×14	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512, C\text{=}32 \\ 1\times1, 1024 \end{bmatrix}\times6$
conv5	7×7	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 1024 \\ 3\times3, 1024, C\text{=}32 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params.		$\mathbf{25.5}\times10^6$	$\mathbf{25.0}\times10^6$
FLOPs		$\mathbf{4.1}\times10^9$	$\mathbf{4.2}\times10^9$

As seen in the table, ResNeXt-50 has 32 as its cardinality is repeated 4 times (depth). The dimensions in denote the residual block structures, whereas the numbers written adjacent to them refer to the number of stacked blocks. 32 precisely denotes that there are 32 groups in the grouped convolution.

3.2.8 Capsule Network

A capsule is a group of neurons that trigger individually for different properties of an object type. Basically, a capsule is a collection of neurons that collectively generate a vector of an action that contains the instantiation value of that neuron with one portion for each neuron. For the most part, the drawbacks of CNNs are linked to the pooling layers. Layers are replaced in Capsule Networks with an acceptable criterion called "routing by agreement". According to the capacity of the child to predict the outputs of the parents, all outputs to all parent capsules are sent on the next sheet. After a few iterations, the results of each parent may overlap with some children's predictions and vary from the others, the parent is present or absent from the scene.

By multiplying a weight matrix by its production, each child computes a prediction vector.

After that, the parent's output is computed as a prediction scalar product with a coefficient reflecting the likelihood that this child belongs to that parent. What predictions of the child are similar to the resulting performance raises the coefficient between those parents?

The following figure 3.2.8 Shows Proposed model Brain tumor classification architecture

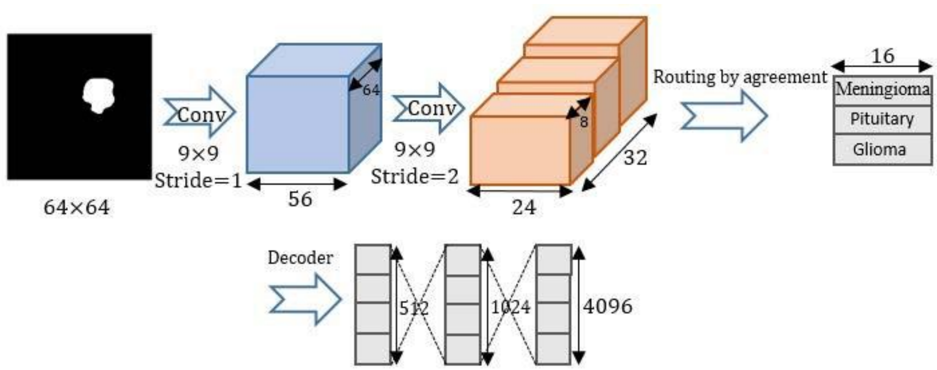


Figure3.2.8: Proposed model Brain tumor classification architecture

Capsules are neuron classes, and different pose parameters are represented by the activity vectors of these neurons. CNNs are often aligned with the layers of pooling. These layers are then replaced by more fitting requirements called “routing by agreement”. In the next sheet, outputs are sent to all parent capsules. Each capsule attempts to predict the outcome. The coupling coefficient increases as this estimate match the real performance of the parent capsule.

Suppose, u_i = output of capsule i. And the parent capsule prediction j.

$$u\hat{\ }j|i = Wijui,$$

Where $u\hat{\ }_{j|i}$ is the vector of prediction of the j^{th} Capsule. W_{ij} this is the Matrix of Weighting. The following Softmax function calculates the coupling coefficients c_{ij}.

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})}$$

b_{ij} is the log probability. At the beginning of the routing according to the agreement process, it is initially set to 0. As follows, the input vector i is determined.

$$sj = \mathrm{X}ciju\hat{\ }j|i.$$

Finally, To minimize capsule production vectors from reaching one, the following non-linear feature for squashing is used to form the final output of each capsule on the basis of its given initial vector value.

$$_j = \frac{\|\boldsymbol{s}_j\|^2}{1 + \|\boldsymbol{s}_j\|^2} \frac{\boldsymbol{s}}{\|\boldsymbol{s}_j\|}$$

The input vector for Capsule j is s_j and the output is v_j. Agreement a_{ij} is then estimated as follows to update log probabilities and coupling coefficients.

$a_{ij} = v_j.\hat{u}_{j|i}$.

A loss function l_k is associated with each capsule k in the last layer. The loss function l_k is determined as follows.

$$l_k = T_k \max(0, m^+ - \|v_k\|)^2 + \lambda(1 - T_k)\max(0, \|v_k\| - m^-)^2$$

Whenever class k is currently present, where T_k is 1 and is 0 otherwise. The hyper parameters are the terms m+, m-, and λ

CHAPTER 4

Experimental result and discussion

4.1 Dataset Description

In this paper, we used the brain MRI dataset [14] to test our proposed architecture. Our dataset consists of a total of 253 images of the brain diagnosed with three brain tumor types. All images are in JPG format. We have used 80% data as a training set and another 20% as a test. It consists of two types of MRI scans:

No-no tumor

YES-cancer

In this dataset, we got 155 images with tumors and 98 images with no tumor. Images have different width and height and different sizes of "black corners". In general, the dimension of the image is (224x224).

All images are in one folder with subfolders yes and no.

4.2 Accuracy Table

Table IV: CapsNet classification test accuracy

Method	Epoch	Accuracy
Baseline	-	81%
CapsuleNet	500	84%
CapsuleNet	1000	91.8%

4.3 Confusion Matrix

Table V: Confusion matrix CapsNet

Actual Class	Predicted Class		
		Tumor	Non-Tumor
	Tumor	143	12
	Non-Tumor	9	89
	Accuracy 91.8%		

CHAPTER 5

Conclusion and Future Scope

5.1 Conclusion

In our paper, we explored the Capsule network usability which is newly proposed to classify the brain tumor type. To detect the exact type of brain tumor from MRI images CapsNets became an improved architecture. Since these networks can operate with fewer training samples, they are way better than CNN. Because of the advanced techniques of CapsNets, CNNs are outperformed in the problem of tumor classification. By adjusting the feature maps numbers in CapsNets convolutional layer, we can also increase the accuracy of brain tumor forms. In addition, the performance of those networks for segmented tumors is better than for all images of the brain, based on our experiments. We intend to explore the impact of providing more layers on the reliability of the classification in the future.

5.2 Future scope

We will try to enhance the architecture's performance to achieve our desired accuracy rate in the future. We have a plan to experiment with a big dataset so that we can get a more accurate result. This time we have used CapsNet for the identification of brain tumors at an early stage but we will try to focus on the classification of brain tumors in our next work. Besides these we will also try to use CapsNets for other fatal disease identification.

REFERENCES

[1] Brain tumor education, available at<<https://www.abta.org/about-brain-tumors/brain-tumor-education/>> Accessed 12 September 2020.

[2] American Cancer Society. *Cancer Facts & Figures 2020*. Atlanta, Ga: American Cancer Society; 2020. Howlader N, Noone AM, Krapcho M, et al (eds). SEER Cancer Statistics Review, 1975-2014, National Cancer Institute. Bethesda, MD, https://seer.cancer.gov/csr/1975_2014/, based on November 2016 SEER data submission, posted to the SEER web site, April 2017.

[3] Statistics adapted from the American Cancer Society's publication, Cancer Facts & Figures 2020, the ACS website (January 2020), the CBTRUS Statistical Report: Primary Brain and Other Central Nervous System Tumors Diagnosed in the United States in 2012-2016 (January 2020), and the National Cancer Institute website (January 2020).

[4] J. Seetha, S. Selvakumar Raja, Biomedical & Pharmacology Journal 11 (3), 1457, 2018, available at<< https://dx.doi.org/10.13005/bpj/1511 >>, Accessed 20 Sept. 2020.

[5] M. Havaei, Axel Davy, David Warde-Farley, Antoine Biard, Aaron Courville, Yoshua Bengio, Chris Palc, Pierre-Marc Jodoin, Hugo Larochelle, "Brain Tumor Segmentation with Deep Neural Networks," Medical Image Analysis, vol.35, pp. 18-31, Jan. 2017.

[6] K. Usman, Kashif Rajpoot, "Brain Tumor Classification from Multi-Modality MRI using Wavelets and Machine Learning," Pattern Analysis and Applications, vol. 20, no. 3, pp.871-881, Aug. 2017.

[7] Jun Cheng, Wei Huang, Shuangliang Cao, Ru Yang, Wei Yang, Zhaoqiang Yun, Zhijian Wang, Qianjin Feng, "Enhanced Performance of Brain Tumor Classification via Tumor Region Augmentation and Partition," PloS one., 2015.

[8] Nidahl K. El Abbadi, Neamah E. Kadhim, "Brain Cancer Classification Based on Features and Artificial Neural Network," International Journal of Advanced Research in Computer and Communication Engineering, vol. 8, no. 1, Jan. 2017.

[9] A. Krizhevsky, I. Sutskever, I. Sutskever, "ImageNet Classification with Deep Convolutional Neural Networks," Neural Information Processing Systems (NIPS) 2012.

[10] J.S. Paul, A.J. Plassard, B.A. Landman, D. Fabbribi, "Deep Learning for Brain Tumor Classification," Proceeings of Spie, 2017.

[11] S. Sabour, N. Frosst, G.E. Hinton, “Dynamic Routing Between Capsules,” 31st Conference on Neural Information Processing Systems (NIPS 2017), 2017.

[12] S Deepak, PM. Ameer, Computers in biology and medicine 111, 103345, 2019. Brain tumor classification using deep CNN features via transfer learning, Sciencedirect, available at<<https://doi.org/10.1016/j.compbiomed.2019.103345>>, Accessed 20 Sept.2020.

[13] Y. Lecun, L. Bottou , Y. Bengio , P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, 1998.

[14] Kaggle, available at <<https://www.kaggle.com/navoneel/brain-mri-images-for-brain-tumor-detection?>>, Accessed 21 Sept. 2020.