Brain Tumor Detection using Convolutional Neural Network

Presented By:

Mohsena Ashraf (15.01.04.012)
Tonmoy Hossain (15.01.04.032)
Fairuz Shadmani Shishir (15.01.04.082)
MD Abdullah Al Nasim (15.01.04.085)

Supervised By:

Mr. Faisal Muhammad Shah
Associate Professor
Department of CSE
Ahsanullah University of Science and Technology.

Introduction

INTRODUCTION

- In the field of Medical Image Analysis, research on Brain tumors is one of the most prominent ones
- Primary brain tumors occur in around 250,000 people a year globally, making up less than 2% of cancers[1]
- Tumor segmentation is one of the most arduous task in medical image
- Classification of the tumor as tumorous or non-tumorous is the primary task

[1]. "Chapter 5.16" *World Cancer Report 2014. World Health Organization. 2014. ISBN 978-9283204299. Archived from the original on 02 May 2019.*

MOTIVATION

- ✓ Well adaptation of automated medical image analysis in the perspective of Bangladesh
- ✓ Early detection of Brain Tumors
- ✓ Reducing the pressure on Human judgement
- ✓ Build a User Interface which can identify the cancerous cells
- ✓ Reducing the death rate by early detection
- ✓ Supporting faster communication, where patient care can be extended to remote areas

CHALLENGES

- Device Independent
- Real-time in erratic background
- Segmenting tumors conjoined with the skull
- Reducing processing time by scaling the hidden layers

Statistics

The following figure shows the net survival rate in case of brain cancer by age for the years 2009-2013.

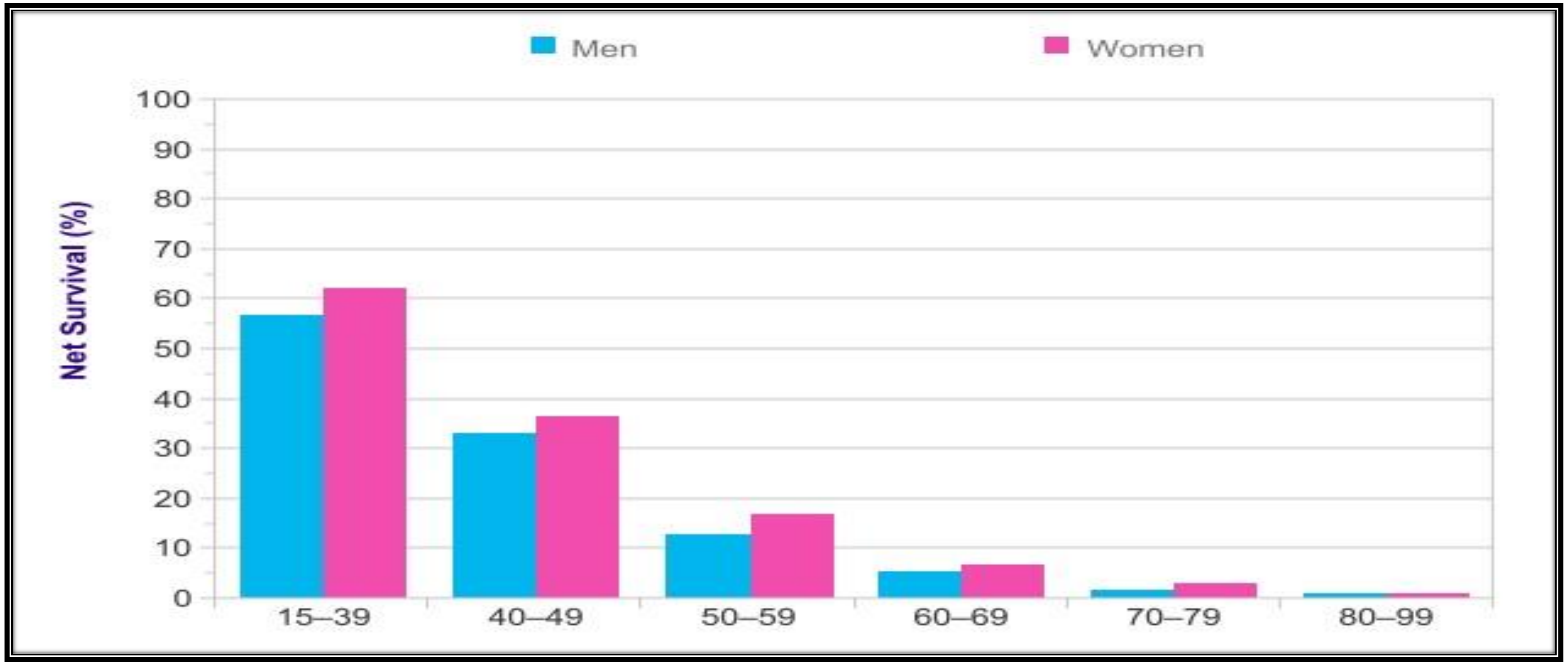


Fig 1: Net survival rate by age for the years 2009-2013 [2]

[2] https://www.cancerresearchuk.org/sites/default/files/cstream-node/surv_5yr_age_brain_0, Last accessed on 15 June, 2019.

The following chart shows the case of brain cancer and percentage of it among all cancers.

Rank	Cancer	New cases diagnosed in 2012 (1,000s)	Percentage among all cancers
2	Brain	256	1.8

Chart 1: Case and Percentage of Brain Tumor [2]

[2] https://www.cancerresearchuk.org/sites/default/files/cstream-node/surv_5yr_age_brain_0, Last accessed on 15 June, 2019.

RESEARCH DOMAIN

Problem

- Segmentation of the tumorous cells
- Detection of the Tumor

- *How can we implement the problem?*
  - Basic Image Processing techniques can be used for segmentation
  - Using Traditional Classifiers
  - Using Convolutional Neural Network based detection

BACKGROUNDS

BRAIN TUMOR

- tumor cells which is undifferentiated in the image
- cells contain abnormal nuclei
- abnormal cells form within the brain
- many dividing cells: disorganized arrangement
- destroy healthy brain cells by invading them
- tumor may grow from neuroma, meningioma, craniopharyngioma or glioma

Types of Brain Tumor

Brain Tumor

Benign

Malignant

Benign

- ✓ non cancerous
- ✓ grows slowly: do not spread into other tissues
- ✓ have clear borders

Malignant

- ✓ brain cancers
- ✓ grows rapidly and invades healthy brain tissues
- ✓ distorted borders

The following figure shows an example of benign and malignant tumor.

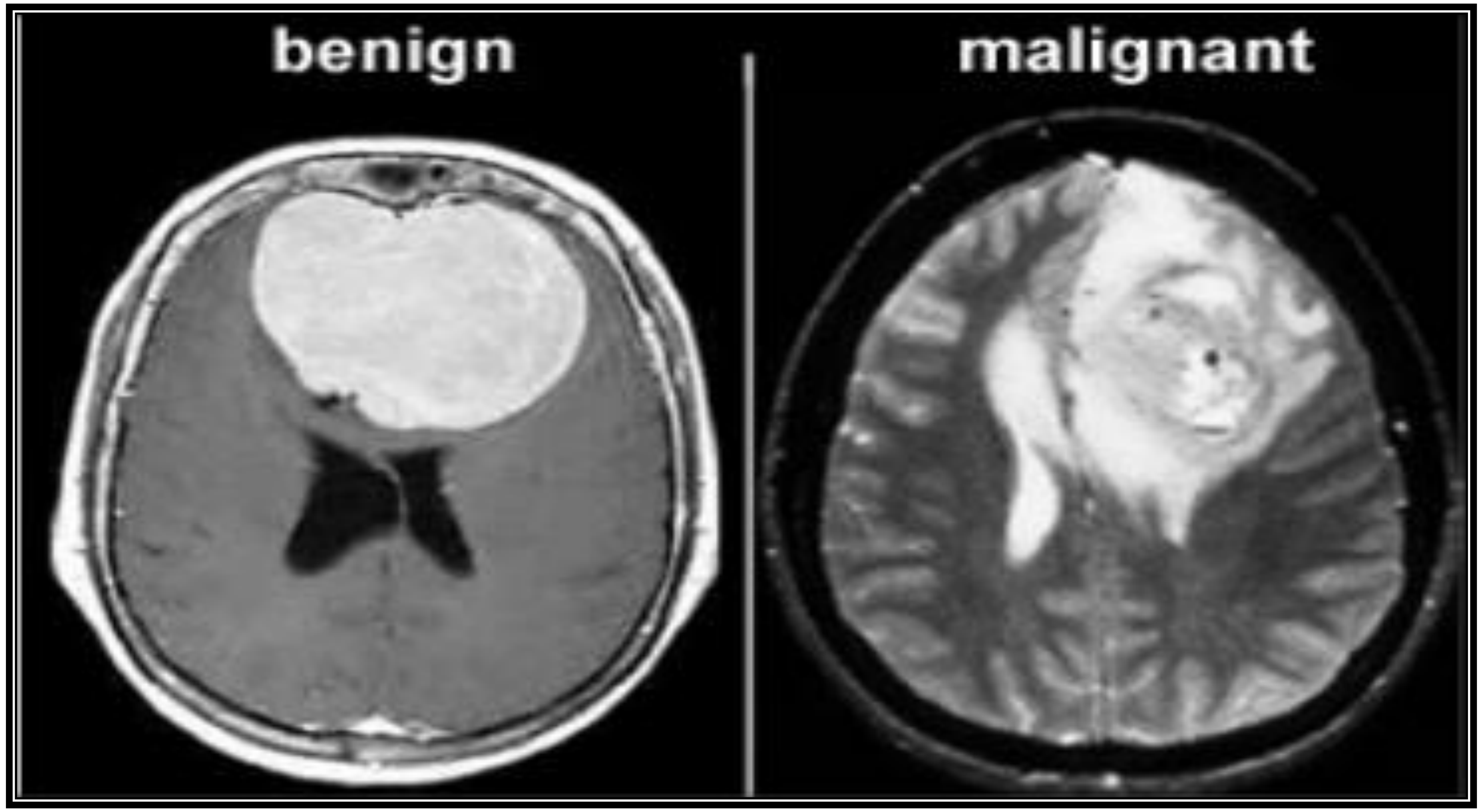


Fig 2: Benign and Malignant Tumor [2]

BACKGROUND STUDIES

Existing Works

- **Devkota et al. 2017**
  - "Image Segmentation for Early Stage Brain Tumor Detection using Mathematical Morphological Reconstruction"
- **Song et al. 2016**
  - "A Novel Brain Tumor Segmentation from Multi-Modality MRI via A Level-Set-Based Model"
- **Dina et al. 2012**
  - "Automated Brain Tumor Detection and Identification using Image Processing and Probabilistic Neural Network Techniques"
- **Zahra et al. 2018**
  - "Brain Tumor Segmentation Using Deep Learning by Type Specific Sorting of Images"

Dataset

Dataset

- BraTS’13 data[3][4]
- Total MRI Image: 217
- Break down intro two category: class-0 and class-1
- All the MRI images are clinically-acquired pre-operative multimodal scans of HGG and LGG
- Described as- T1, T1Gd, T2 and FLAIR volumes

- Some Examples

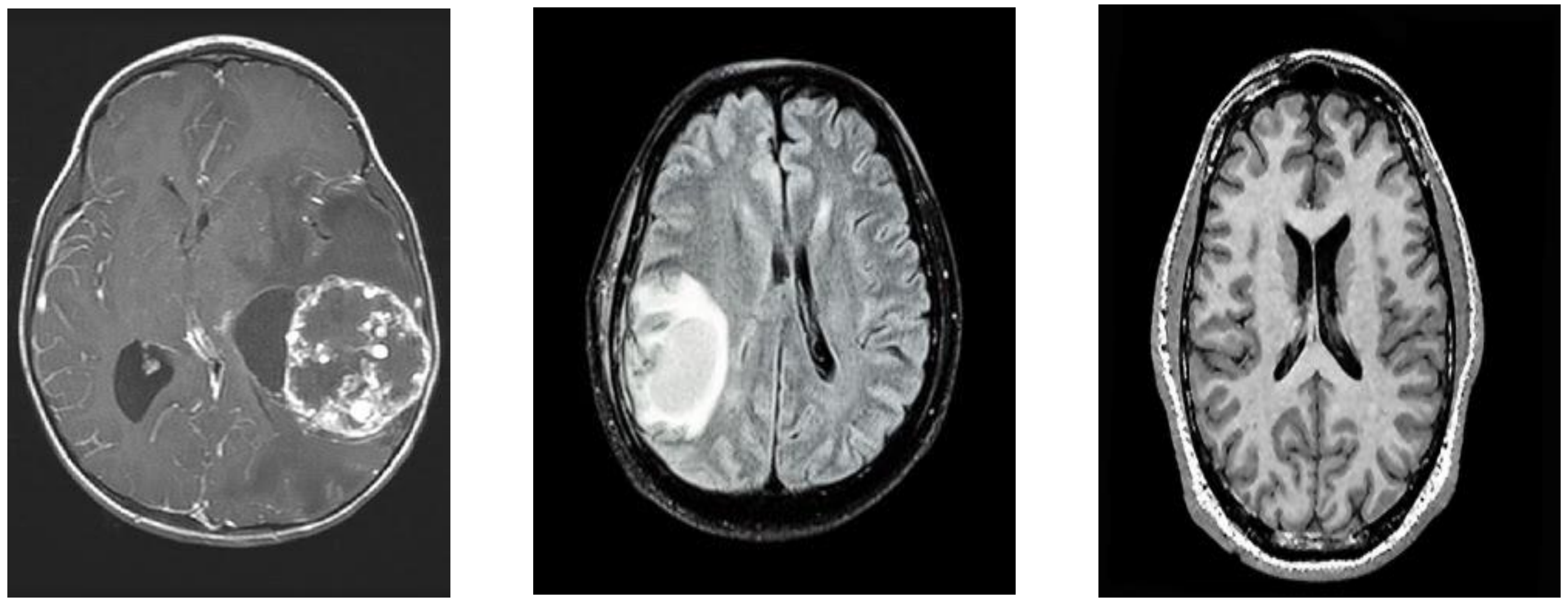


Fig 3: Some example of Dataset[3][4]

[3] Menze BH, Jakab A, Bauer S, Kalpathy-Cramer J, Farahani K, Kirby J, Burren Y, Porz N, Slotboom J, Wiest R, Lanczi L, Gerstner E, Weber MA, Arbel T, Avants BB, Ayache N, Buendia P, Collins DL, Cordier N, Corso JJ, Criminisi A, Das T, Delingette H, Demiralp Г, Durst CR, Dojat M, Doyle S, Festa J, Forbes F, Geremia E, Glocker B, Golland P, Guo X, Hamamci A, Iftekharuddin KM, Jena R, John NM, Konukoglu E, Lashkari D, Mariz JA, Meier R, Pereira S, Precup D, Price SJ, Raviv TR, Reza SM, Ryan M, Sarikaya D, Schwartz L, Shin HC, Shotton J, Silva CA, Sousa N, Subbanna NK, Szekely G, Taylor TJ, Thomas OM, Tustison NJ, Unal G, Vasseur F, Wintermark M, Ye DH, Zhao L, Zhao B, Zikic D, Prastawa M, Reyes M, Van Leemput K. "The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS)", IEEE Transactions on Medical Imaging 34(10), 1993-2024 (2015) DOI: 10.1109/TMI.2014.2377694

[4] Bakas S, Akbari H, Sotiras A, Bilello M, Rozycki M, Kirby JS, Freymann JB, Farahani K, Davatzikos C. "Advancing The Cancer Genome Atlas glioma MRI collections with expert segmentation labels and radiomic features", Nature Scientific Data, 4:170117 (2017) DOI: 10.1038/sdata.2017.117

METHODOLOGY (Traditional Machine Learning)

✓ Proposed Method for tumor segmentation and classification using traditional classifiers

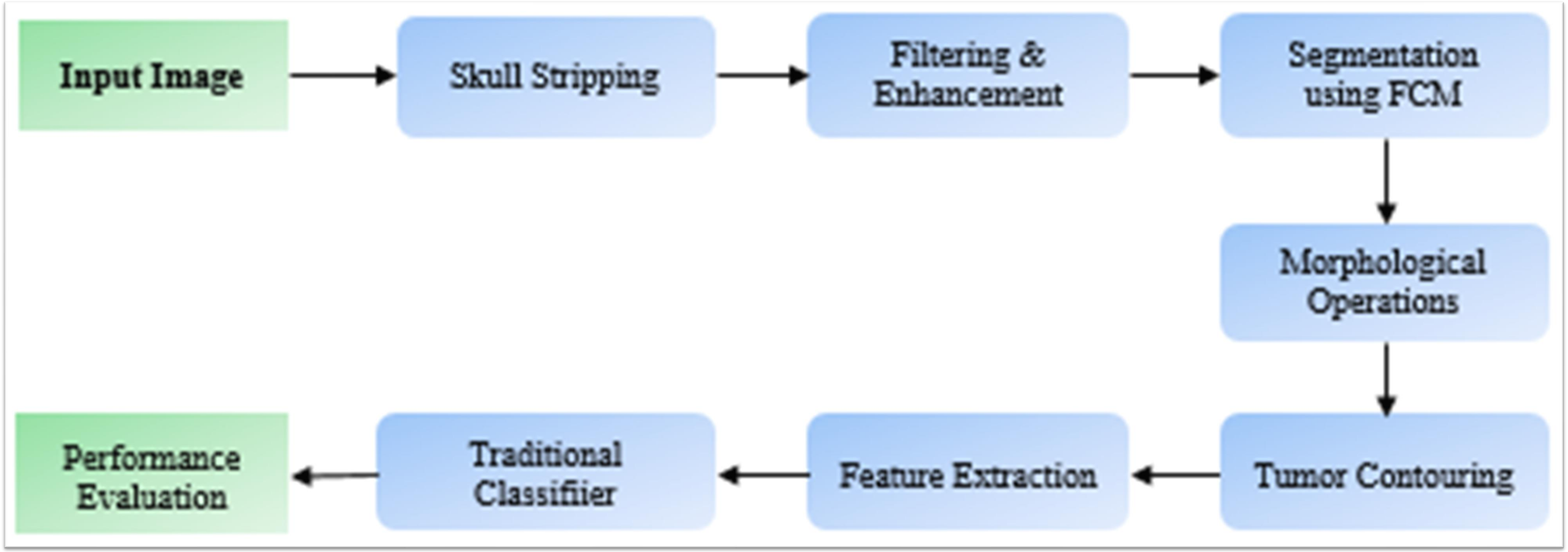


Fig 4: Proposed methodology for classification using Traditional Classifiers

✓ Elaborated proposed methodology for segmentation and classification using traditional Machine learning techniques

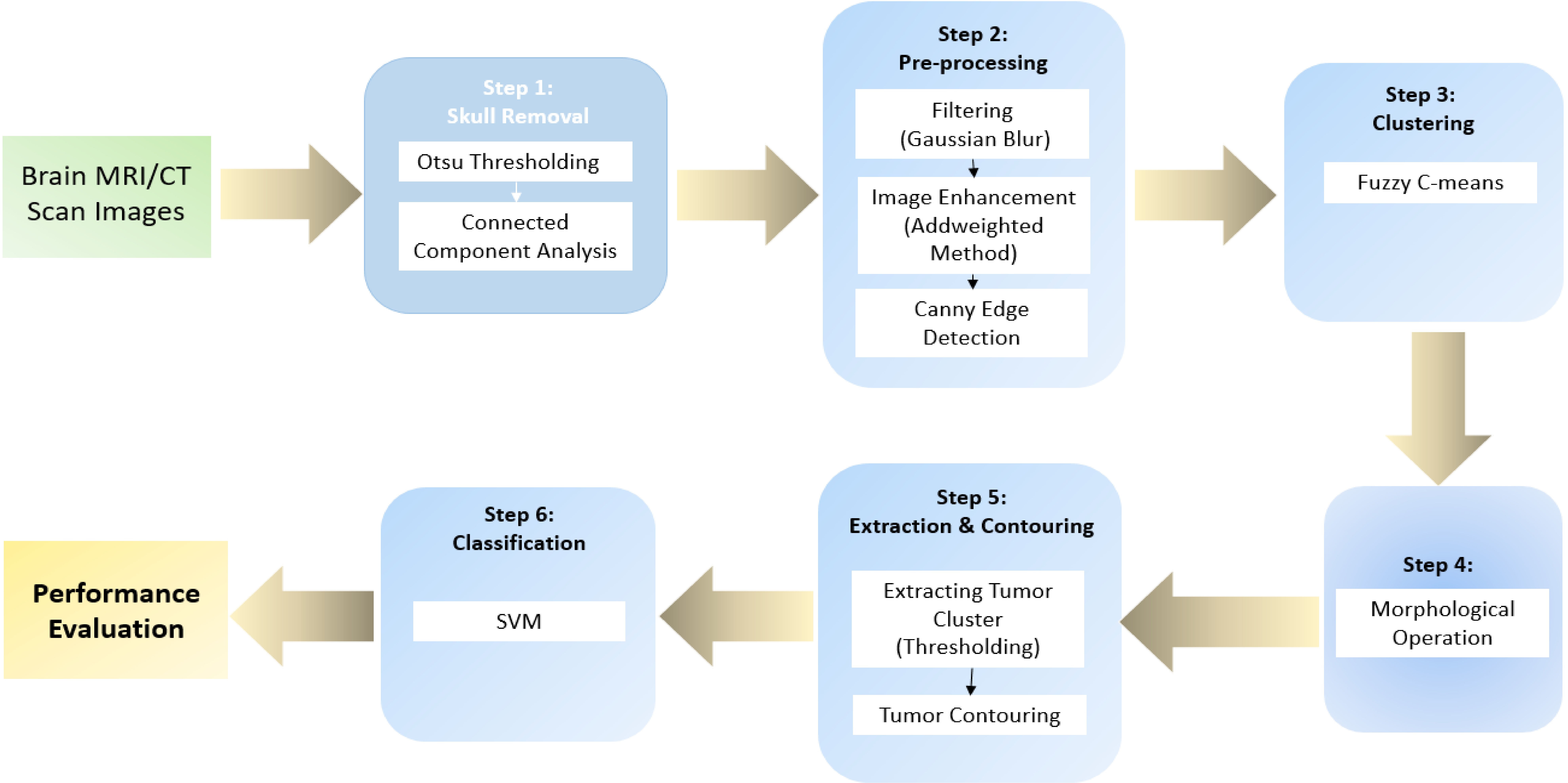


Fig 5: elaborated proposed methodology

Stage-1:Skull Stripping

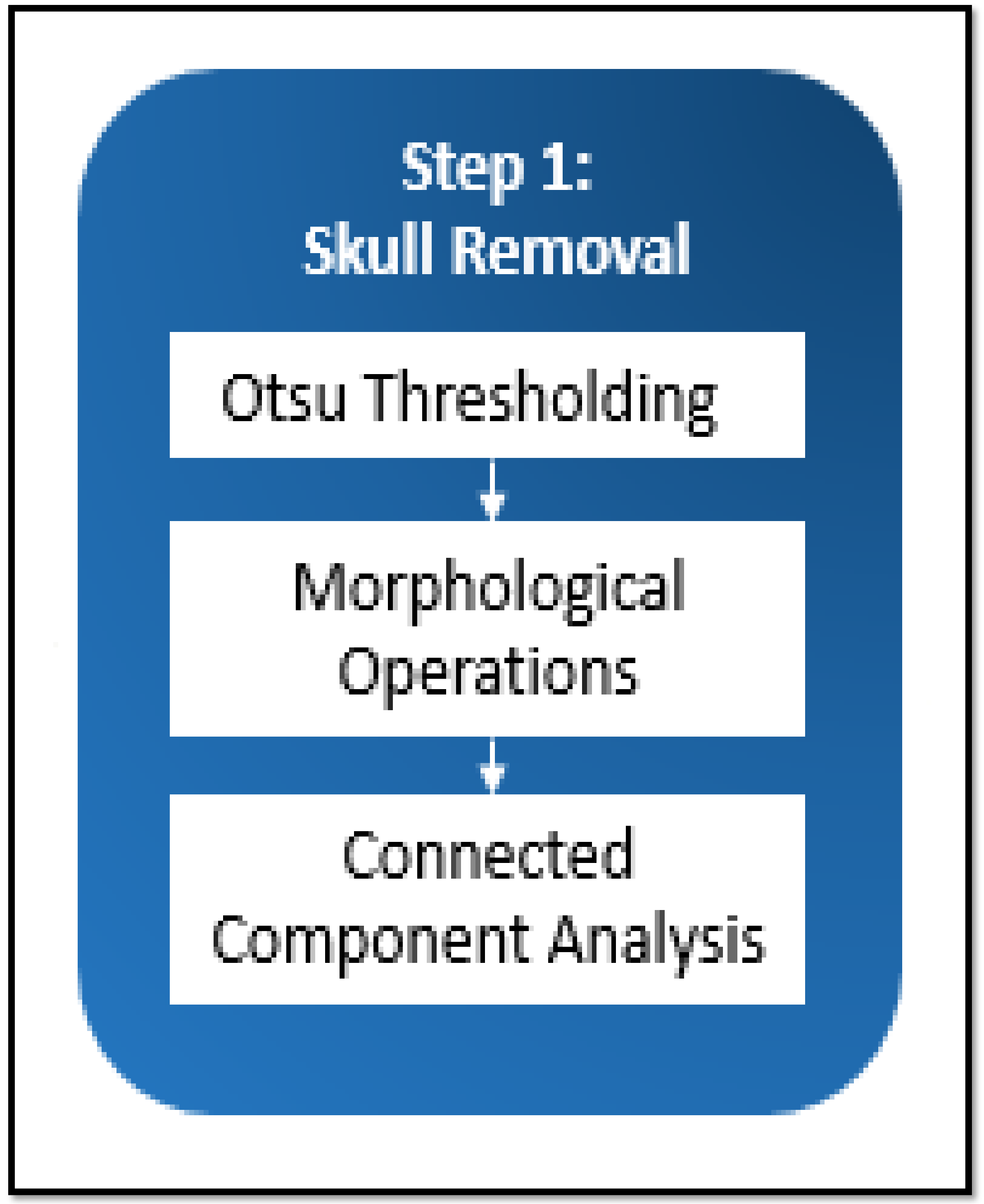


Fig 6: process of skull removal

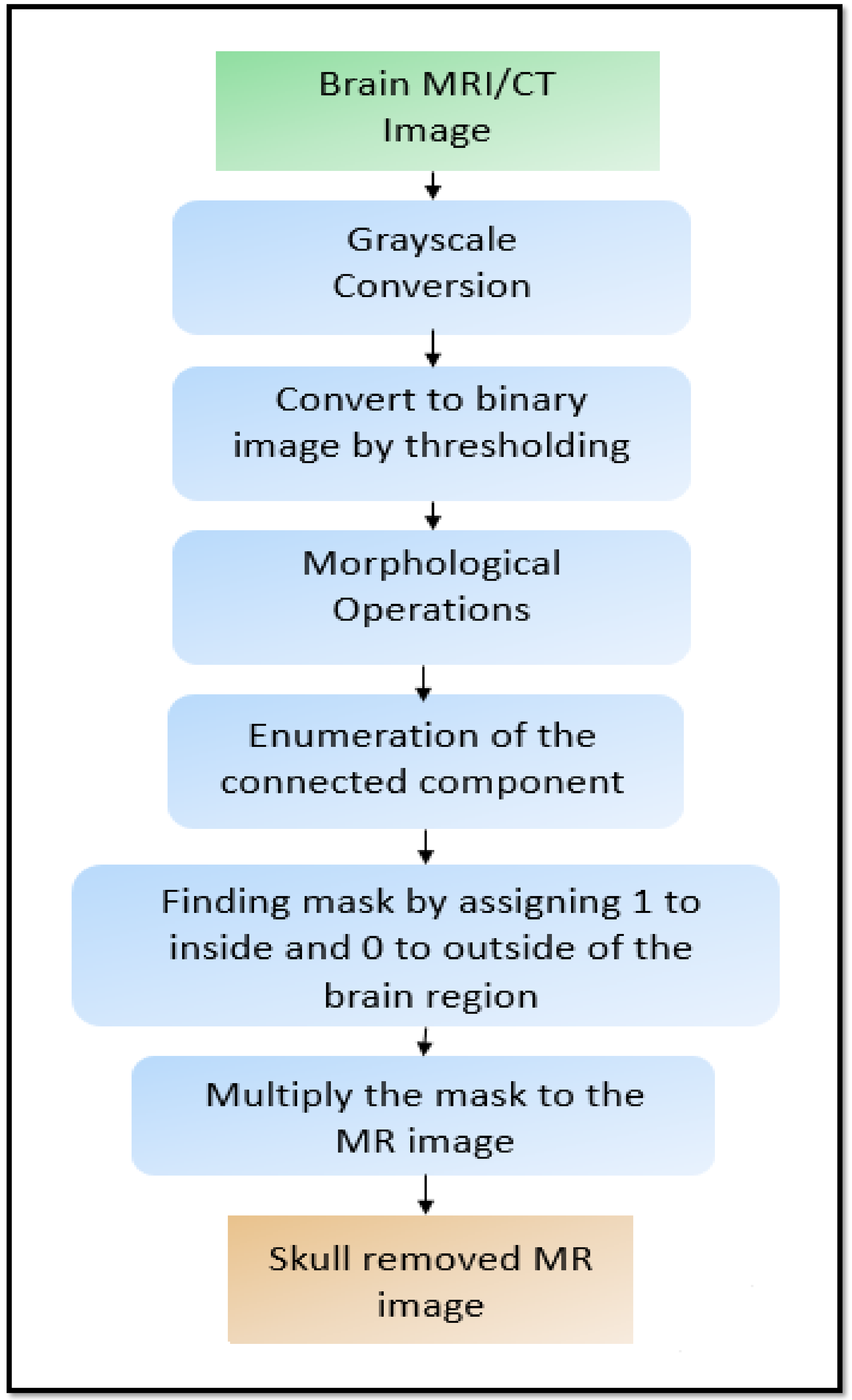


Fig 7: elaborated process of skull removal

Stage-1:Skull Stripping

- Converted our MRI Images into Grayscale
- OTSU Thresholding was applied for binarization
- Erosion operation had been performed before applying connected component analysis
- Each maximal region of connected pixels (not separated by boundary) is called a connected component. We found the largest component which is the skull
- We found the mask by assigning 1 to inside and 0 to outside of the brain region
- Multiplied the mask to T1, T2 and FLAIR images

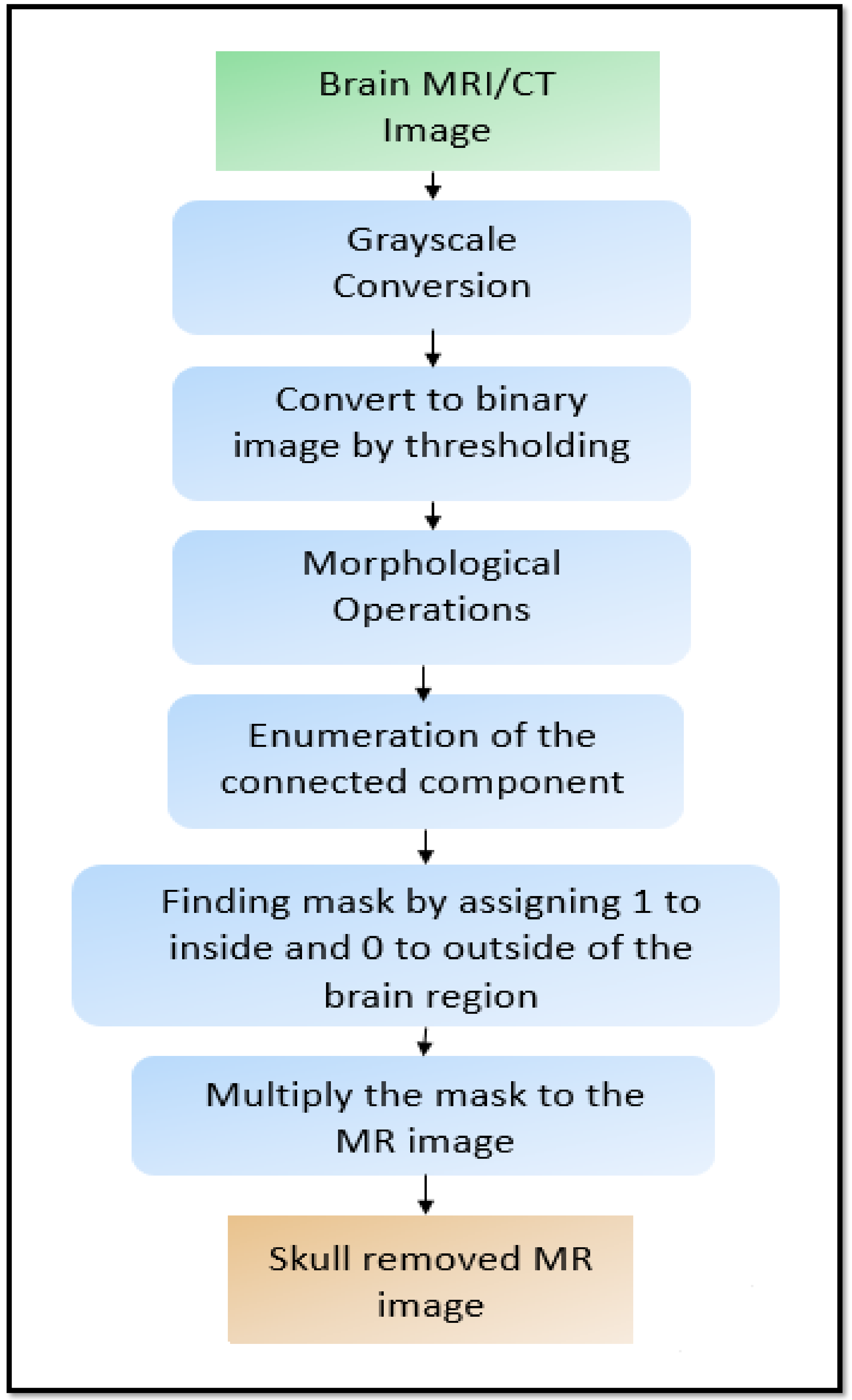


Fig 8: elaborated process of skull removal

Stage-1:Skull Stripping

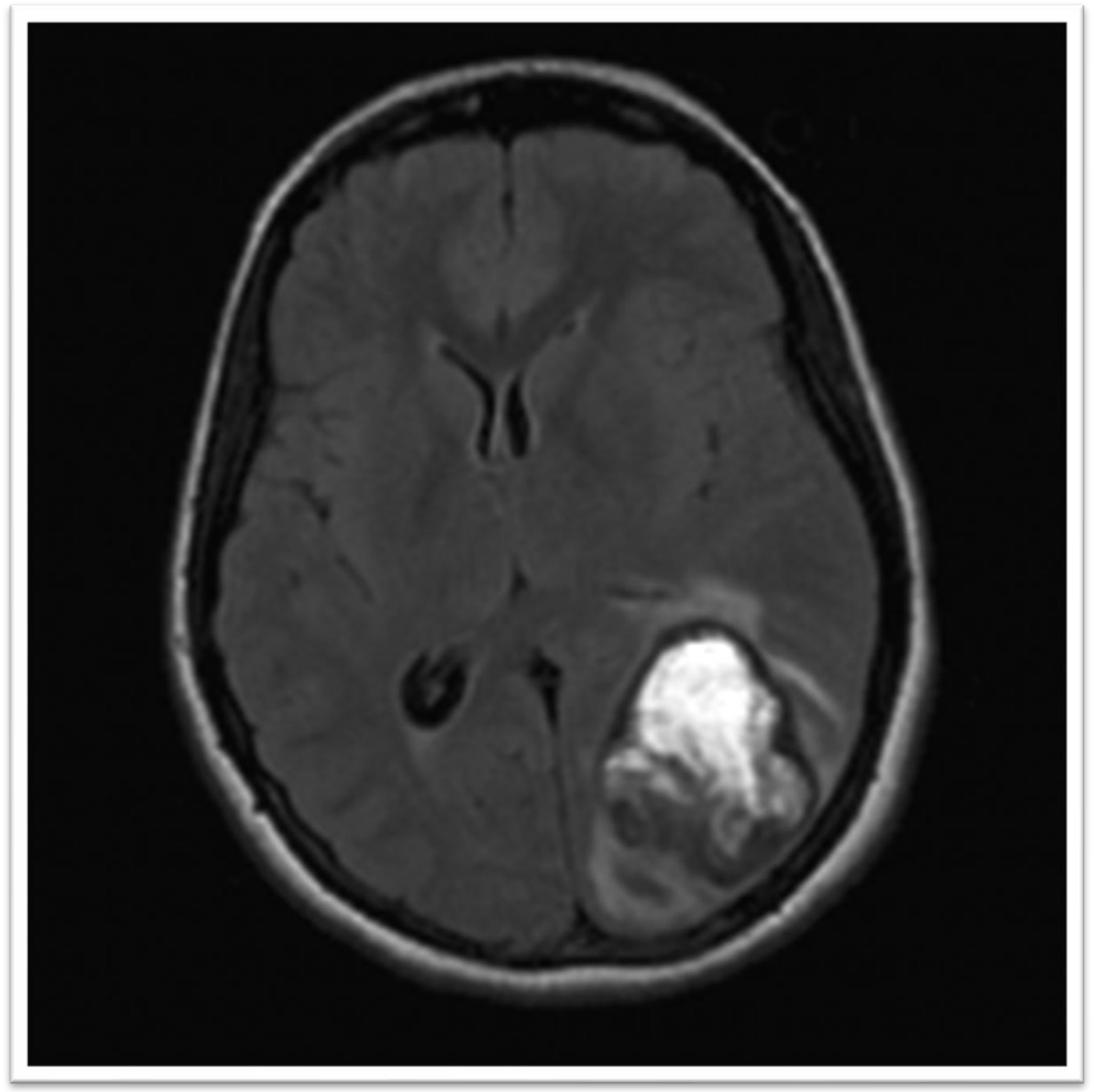


Figure 9.1: input image

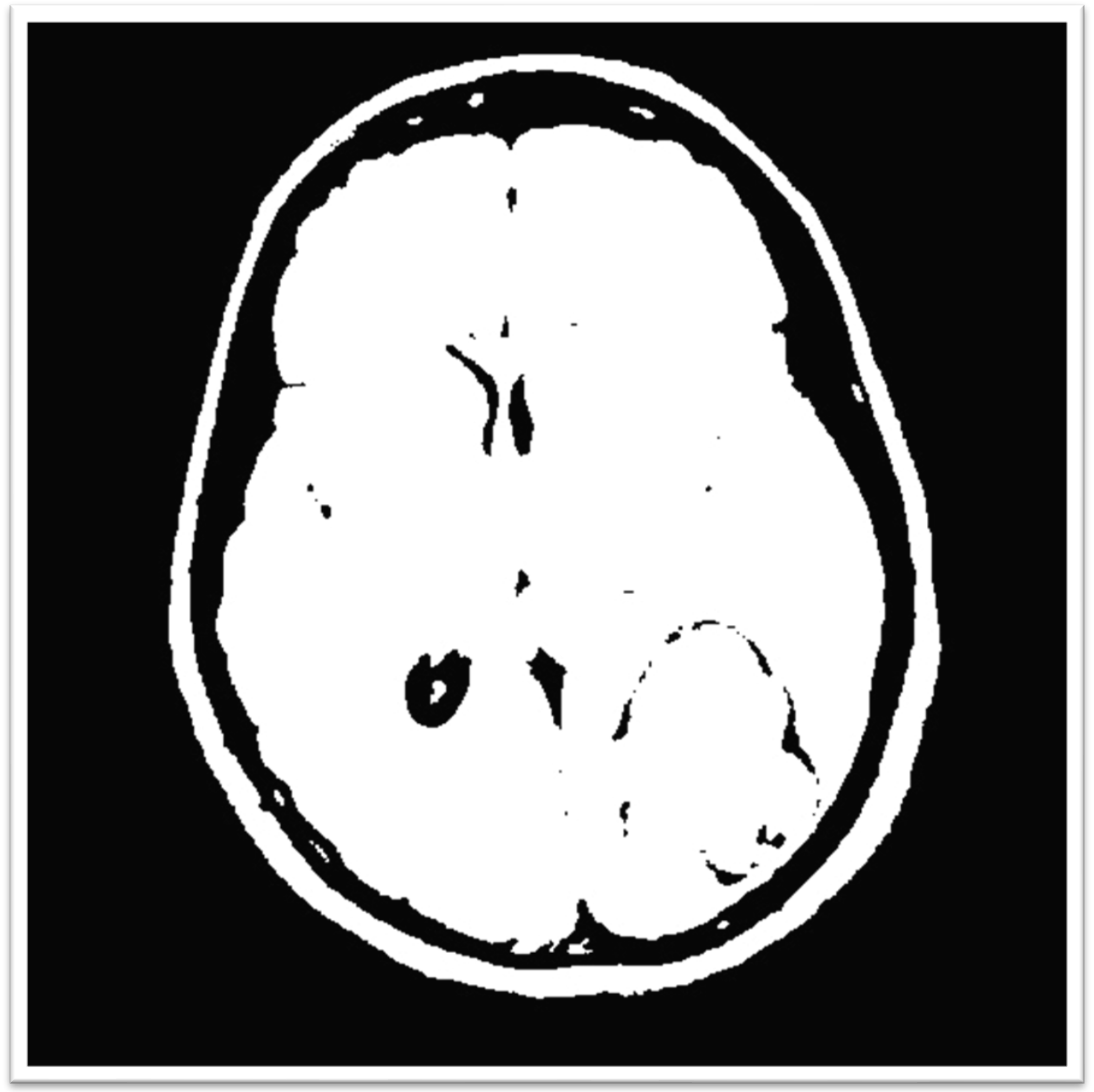


Fig 9.2: thresholded image

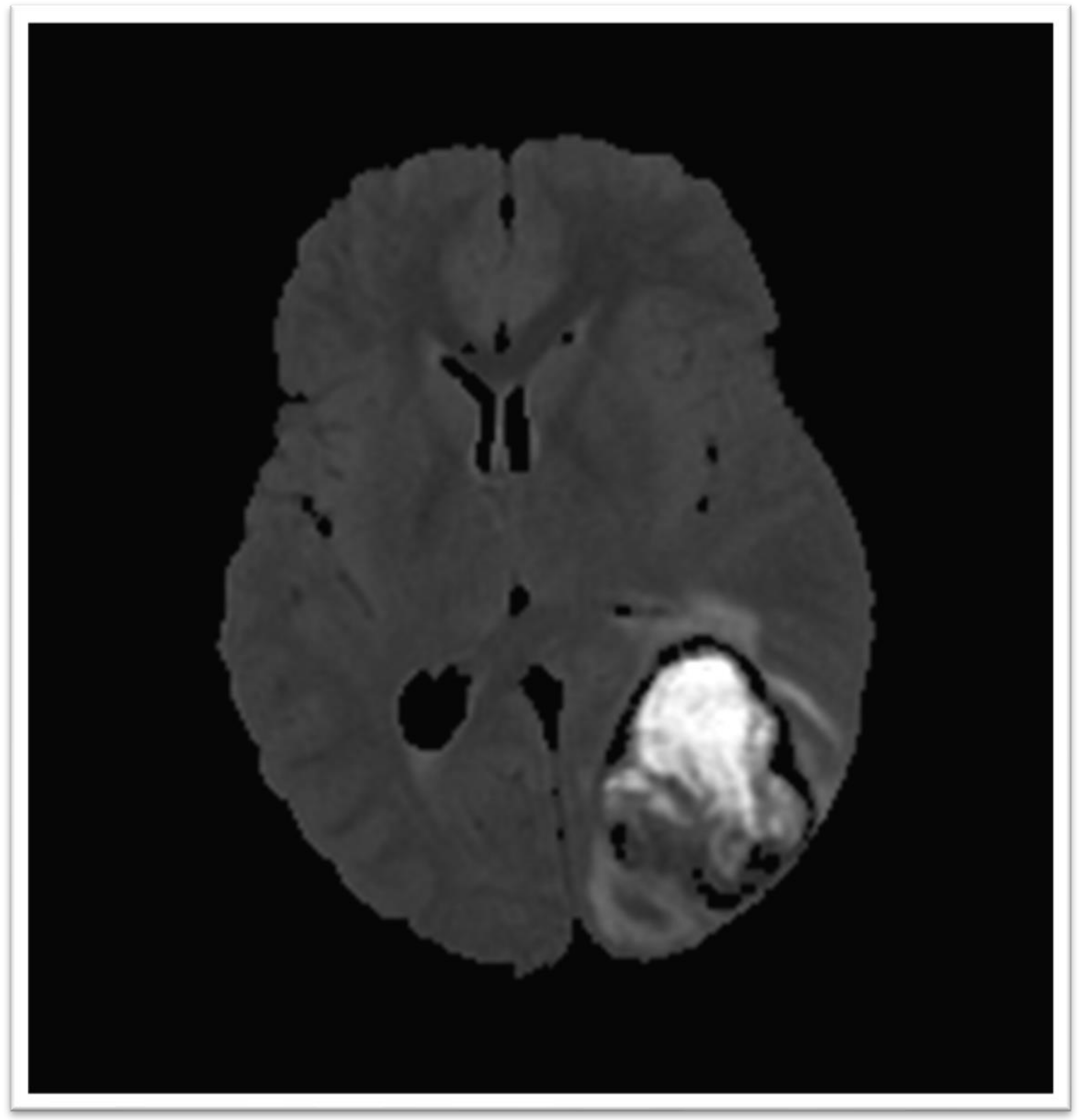
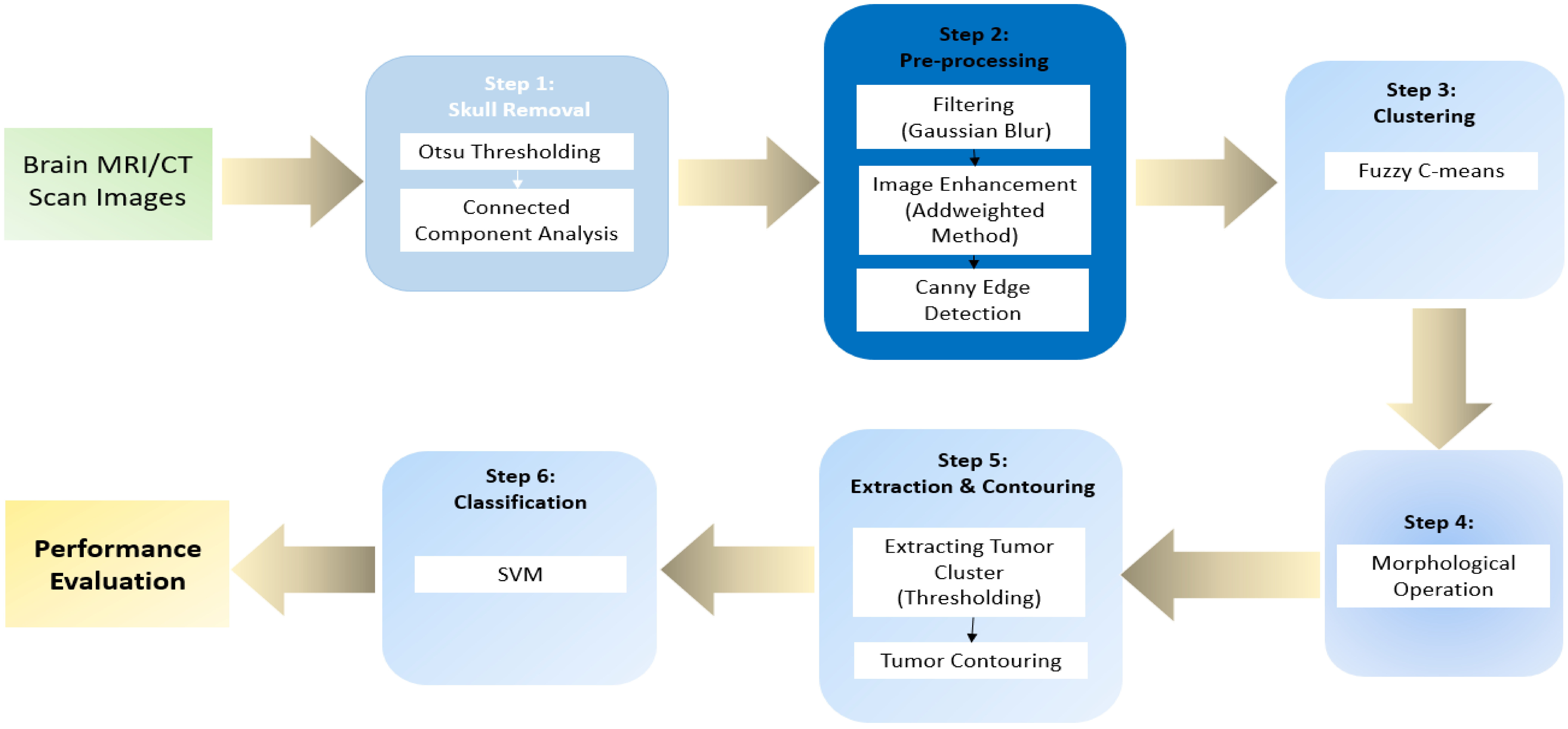


Fig 9.3: skull removed image

Fig 9: steps of skull stripping

Stage-2: Pre-Processing



Stage-2: Pre-Processing

- Median filter gives us the most prominent result for noise removal
- For enhancing the image quality, we used the add-weighted method
  - Blur the image
  - Subtract the blurred image from the original image
  - Output image will have most of the high-frequency components
- Applied the Canny Edge Detection method for detecting the edges

Stage-2: Pre-Processing

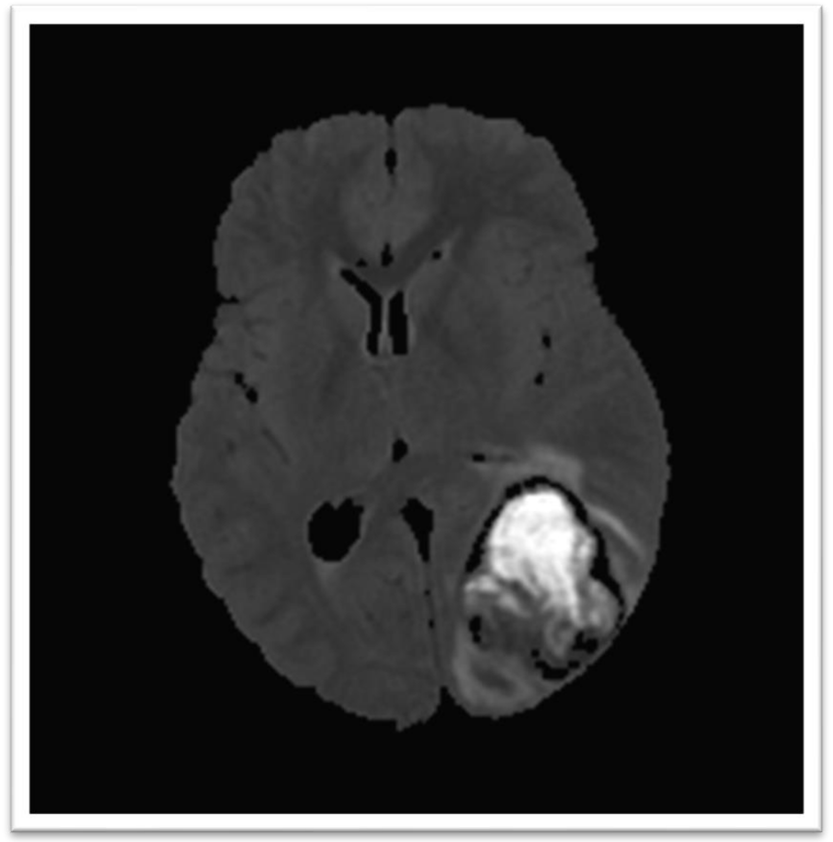


Fig 10.1(a): skull removed MRI

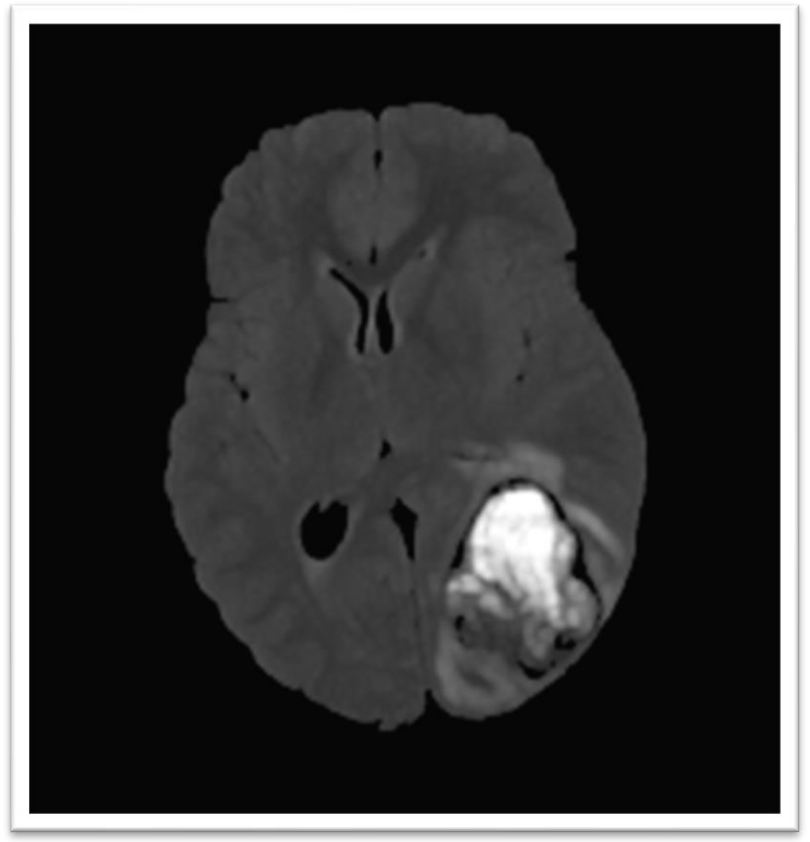


Fig 10.1(b): gaussian blur filter

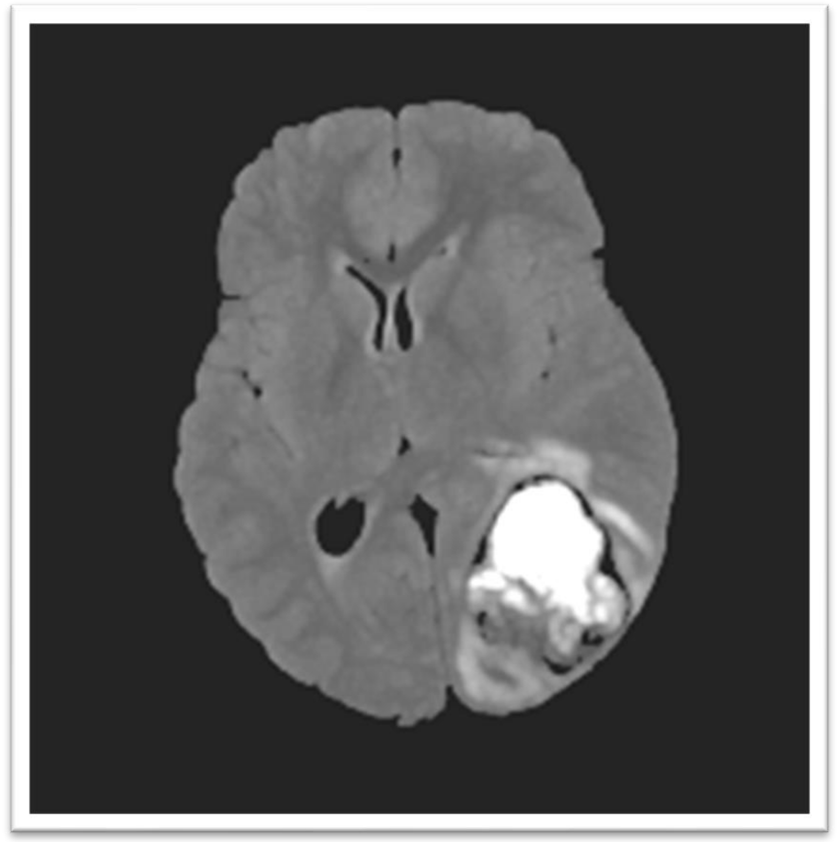


Fig 10.1(c): enhanced MRI

Fig 10.1(d): edge detection MRI

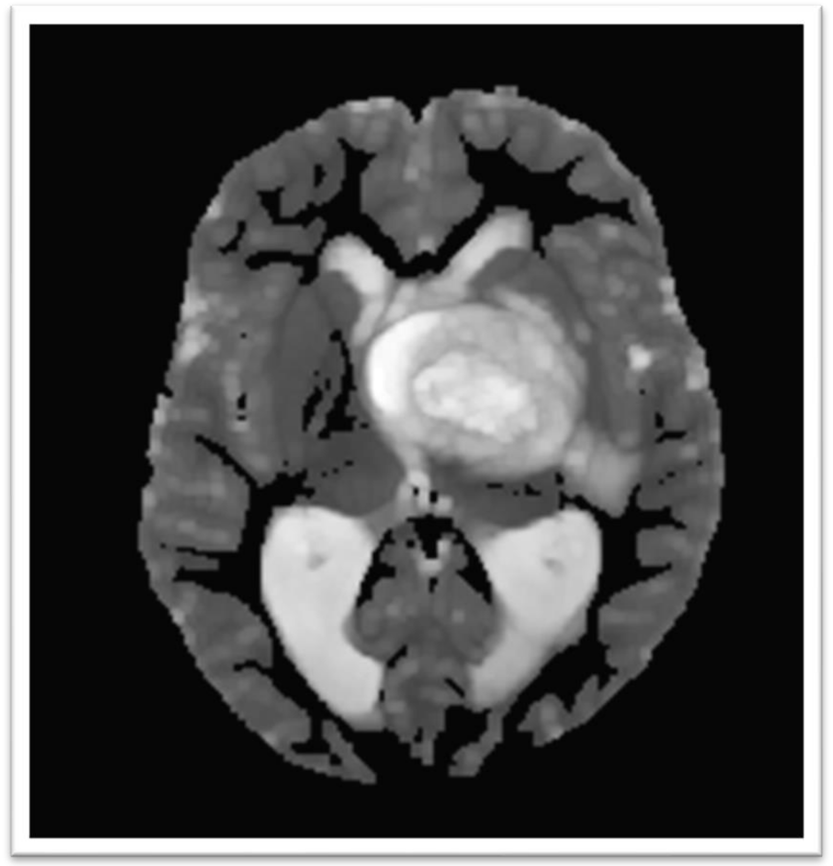


Fig 10.2(a): skull removed MRI

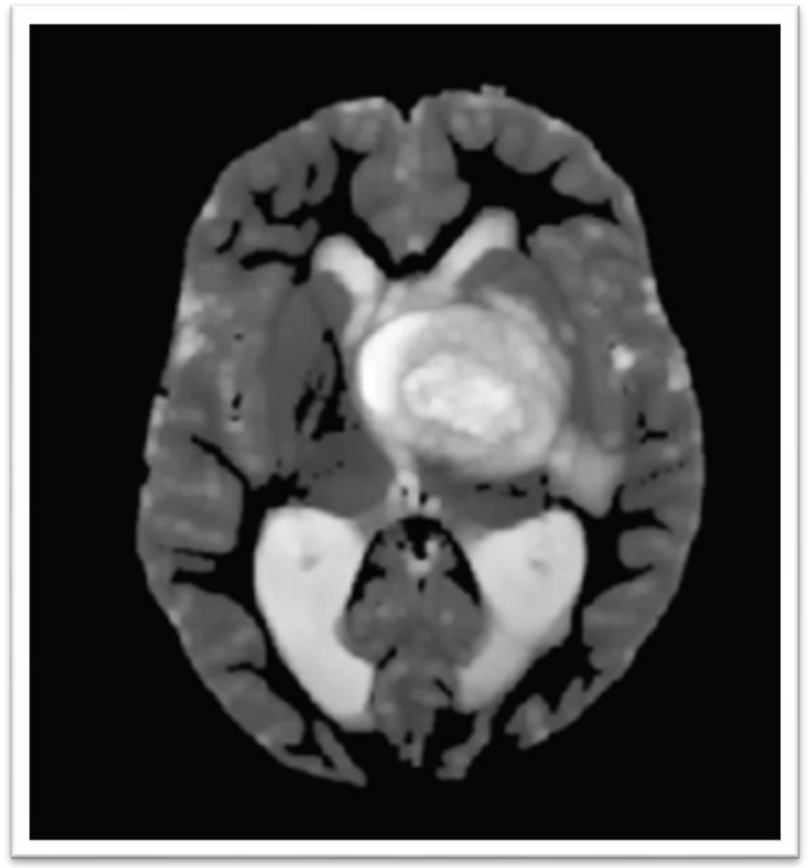


Fig 10.2(b): gaussian blur filter

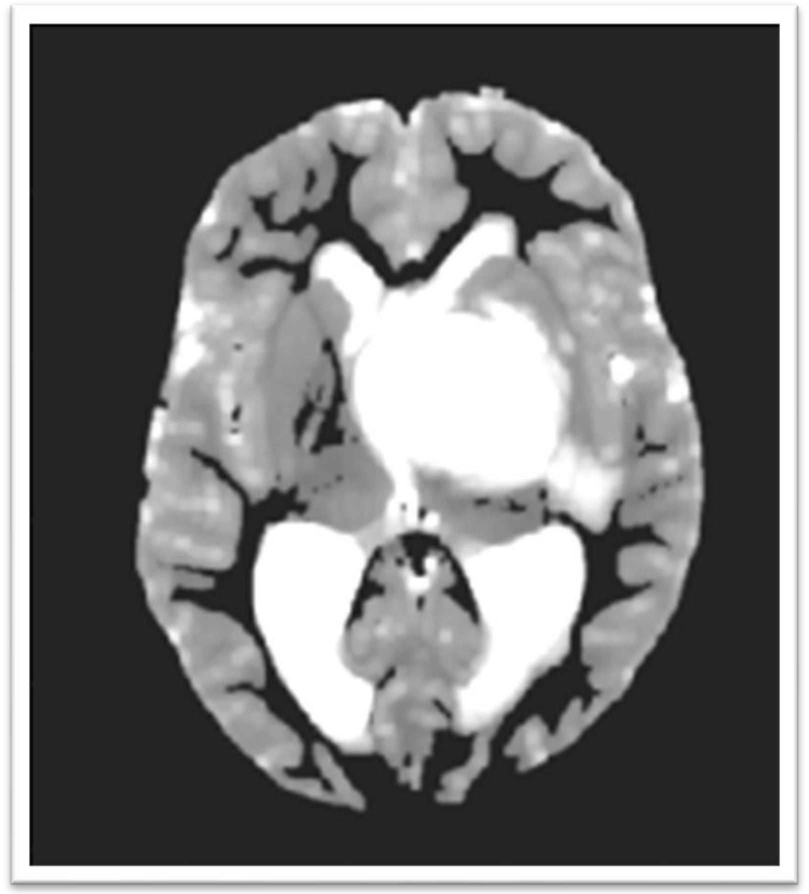
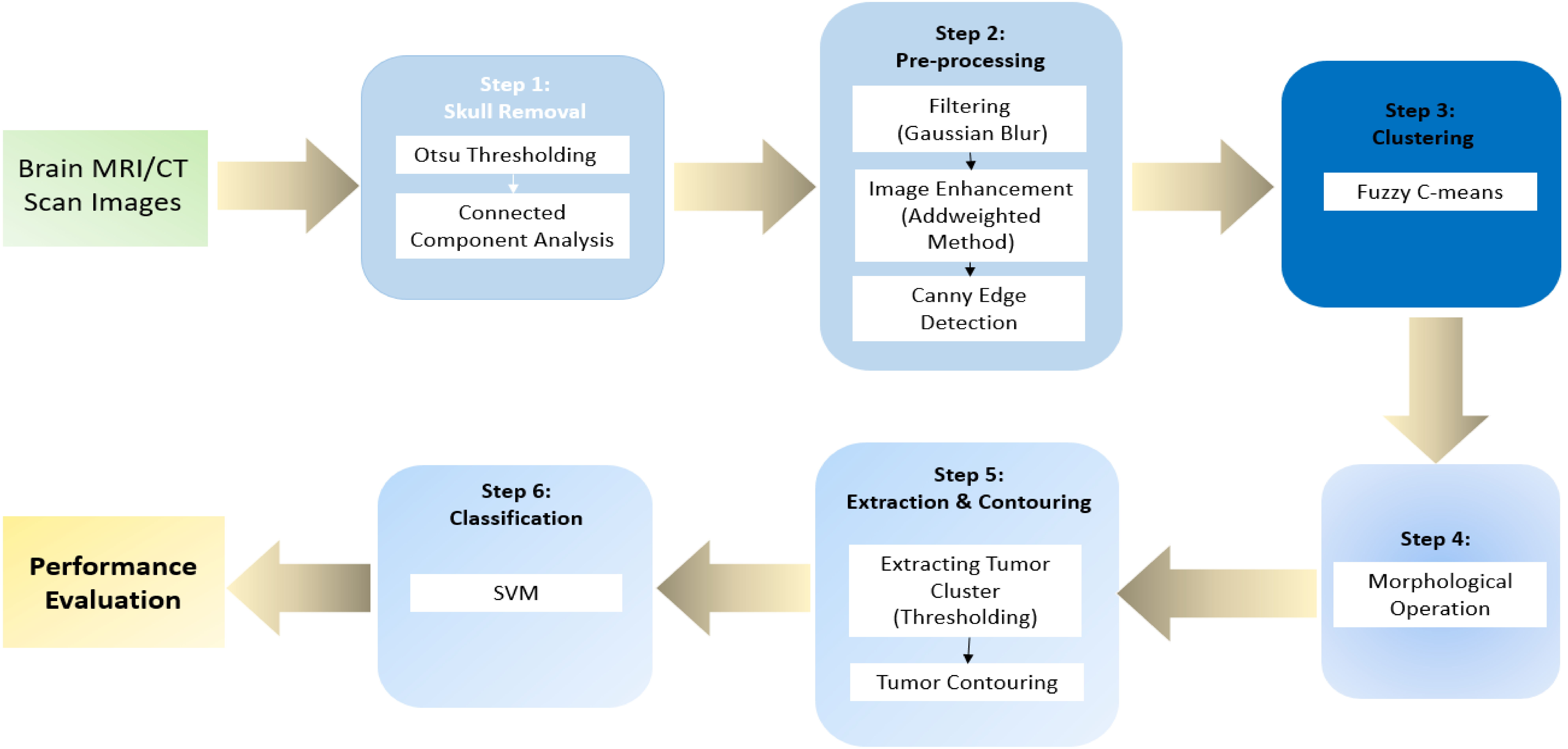


Fig 10.2(c): enhanced MRI

Fig 10.2(d): edge detection MRI

Fig 10: steps of pre processing of the image

Stage-3: Clustering



Stage-4: Clustering

- **Segmentation Using Fuzzy C-Means (FCM)**
  - ✓ A method of clustering which allows one piece of data to belong to two or more clusters
  - ✓ Involves assigning data points to clusters
  - ✓ Items in the same cluster are as similar as possible
  - ✓ Items belonging to different clusters are as dissimilar as possible

Segmentation Using FCM

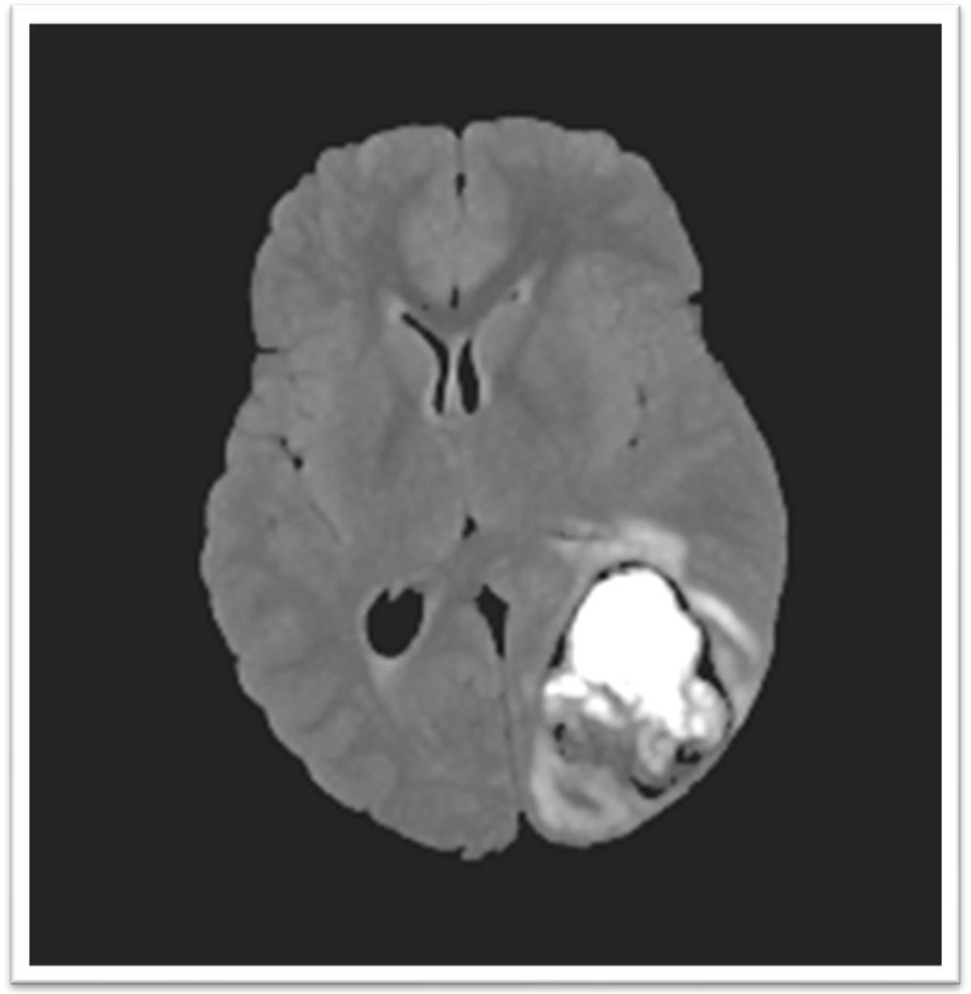


Fig 11.1(a): enhanced MRI

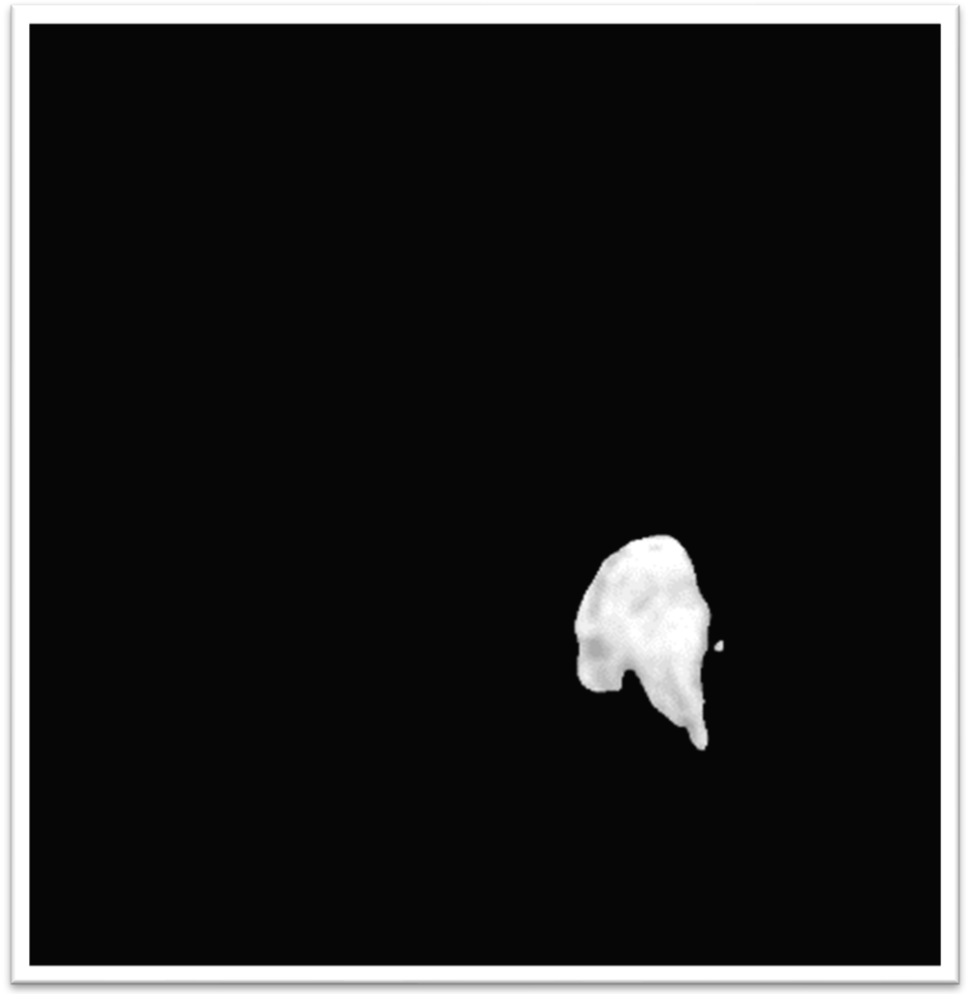


Fig 11.1(b): segmented tumor

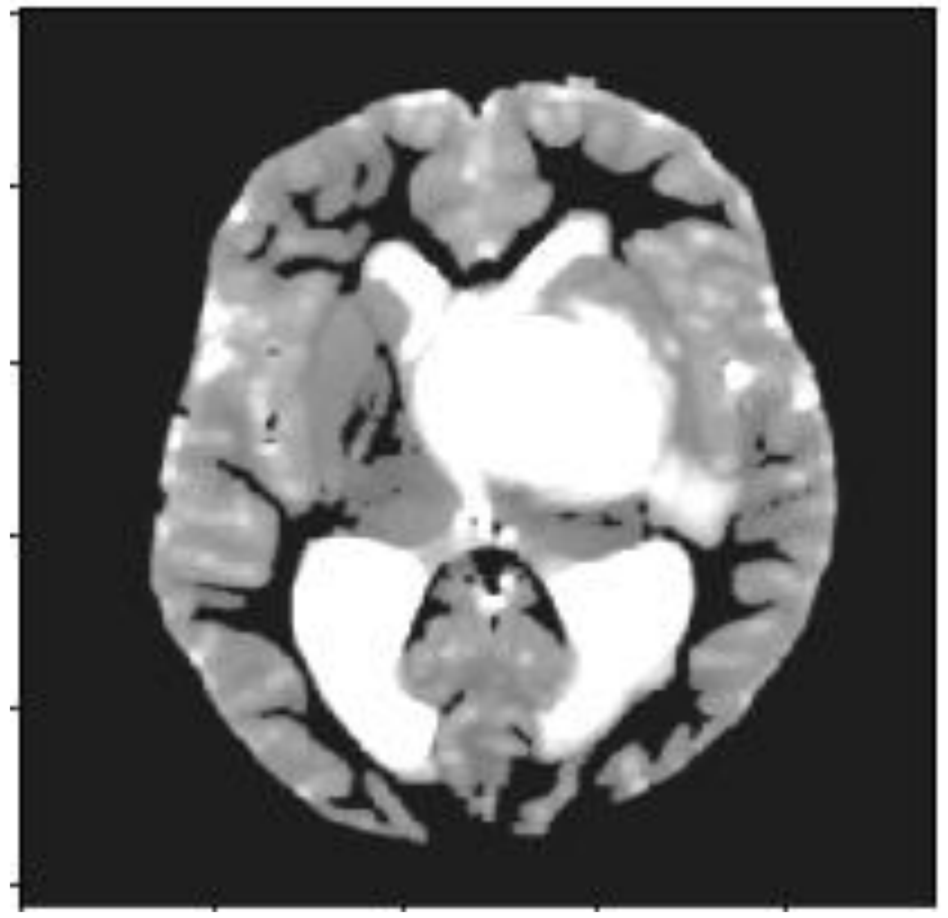


Fig 11.2(a): enhanced MRI

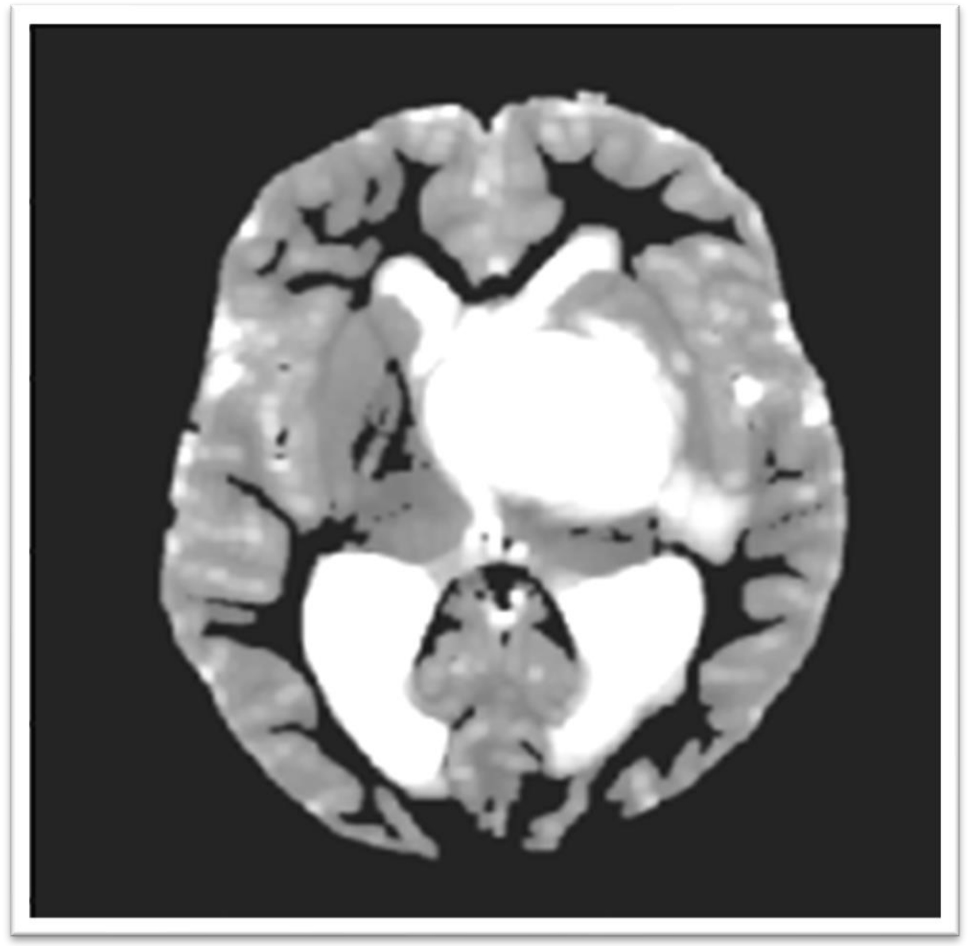


Fig 11.2(b): segmented tumor

Fig 11: segmentation using FCM

Segmentation Using K-Means Clustering

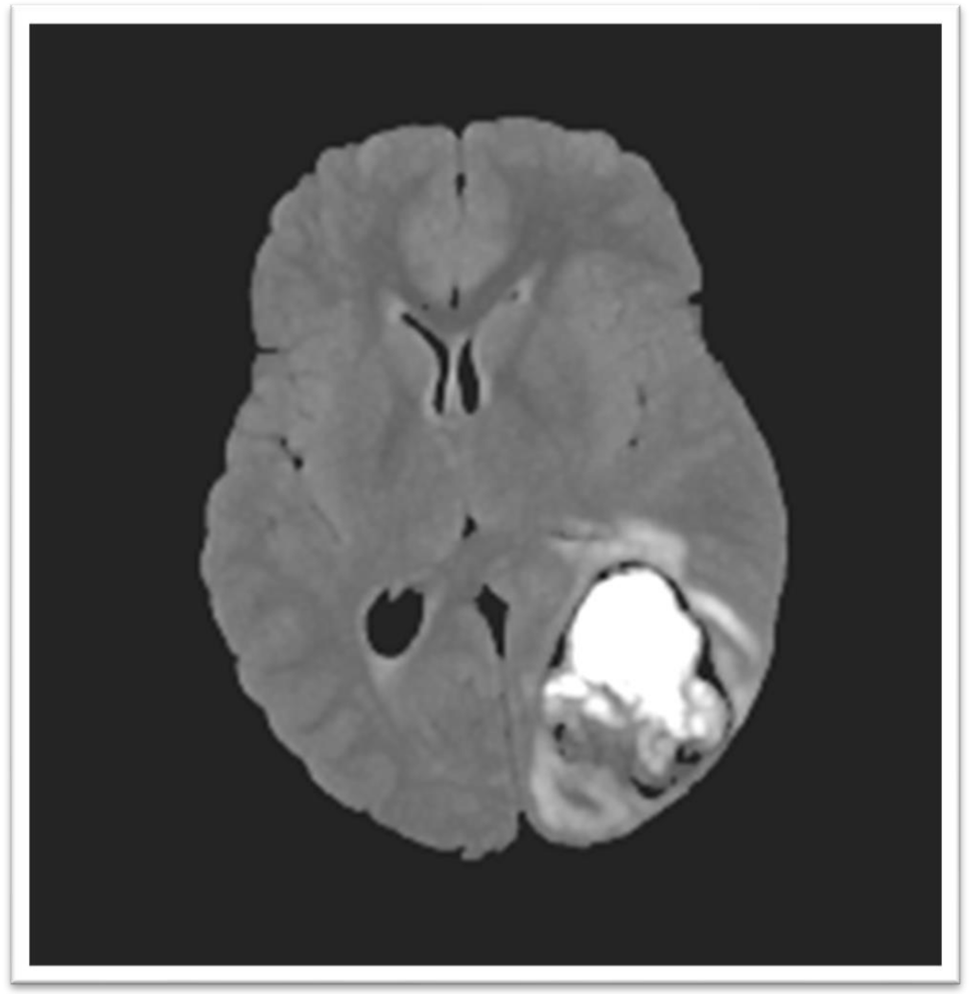


Fig 12.1(a): enhanced MRI

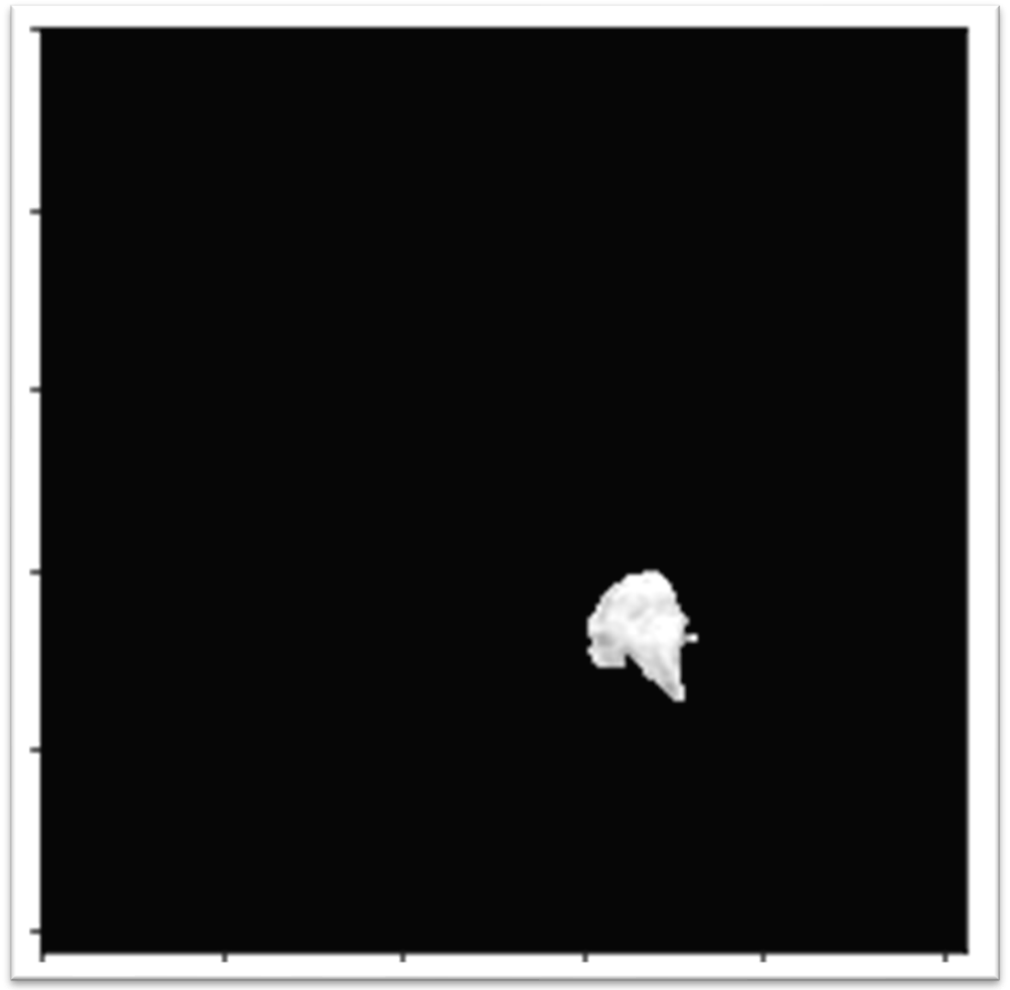


Fig 12.1(b): segmented tumor

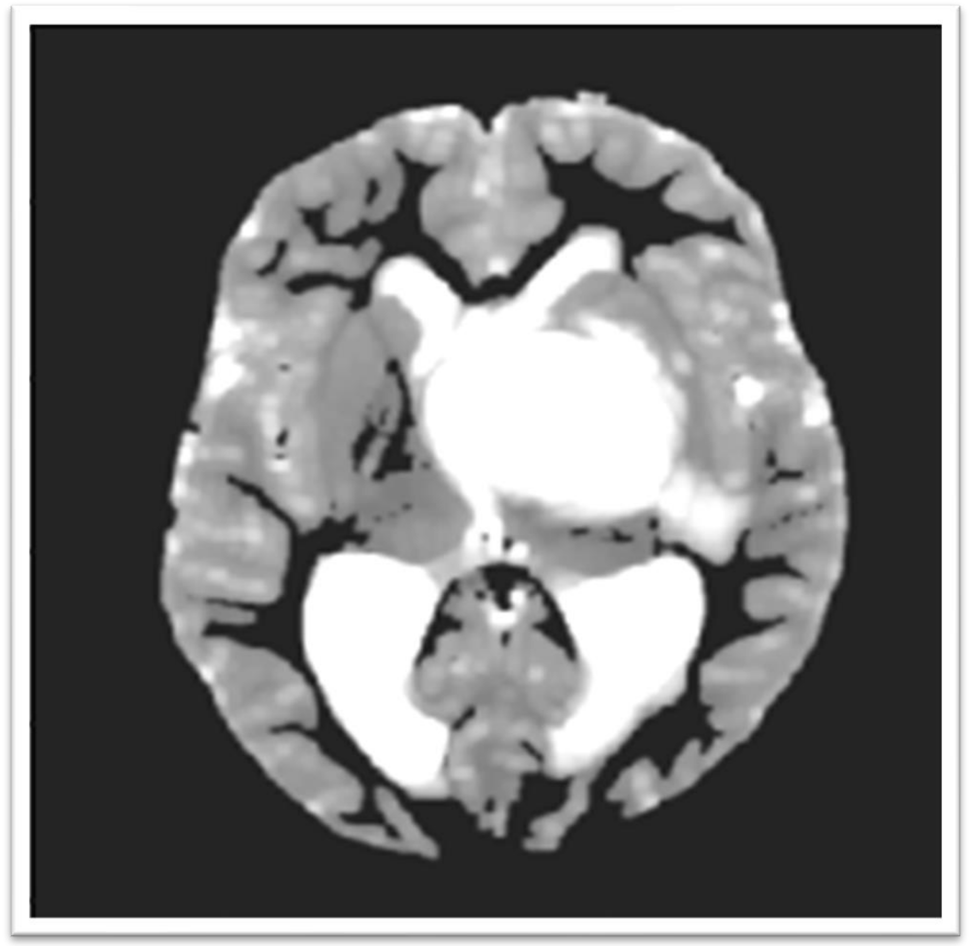


Fig 12.2(a): enhanced MRI

Fig 12.2(b): segmented tumor

Fig 12: segmentation using K-Means Clustering

Segmentation Using Watershed Algorithm

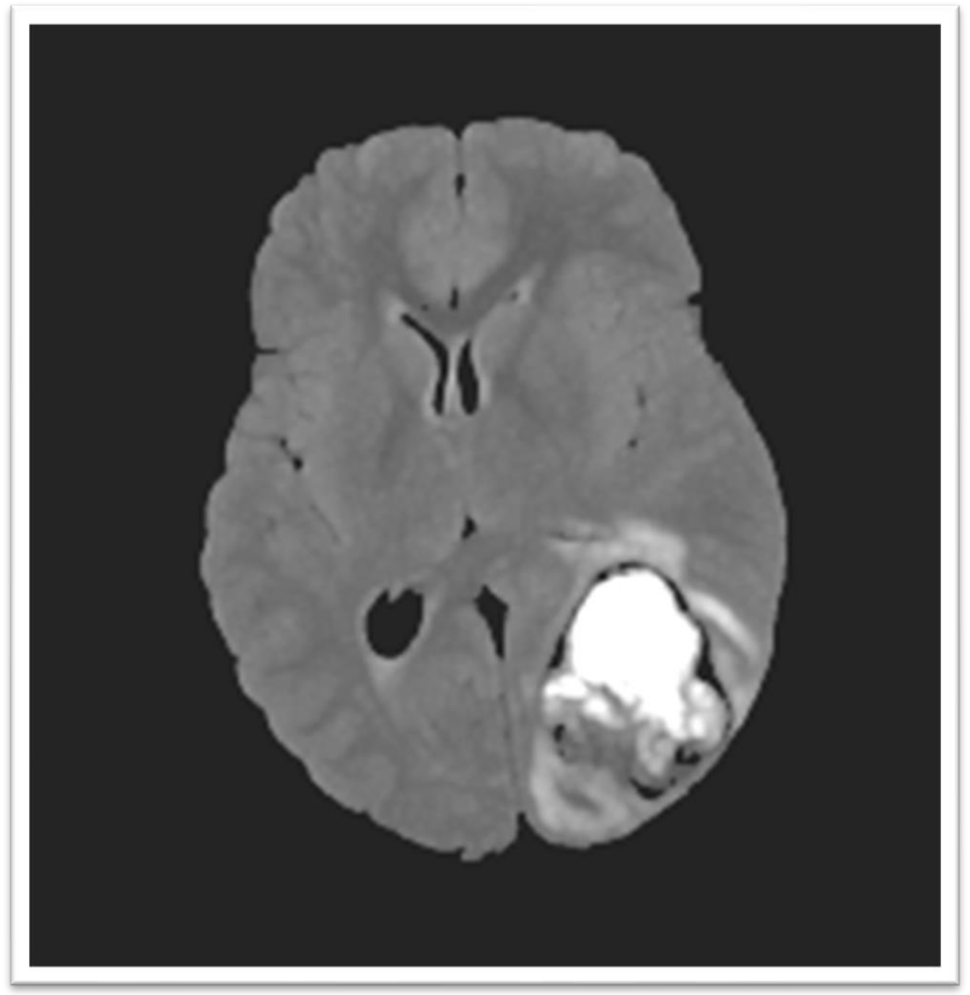


Fig 13.1(a): enhanced MRI

Fig 13.1(b): segmented tumor

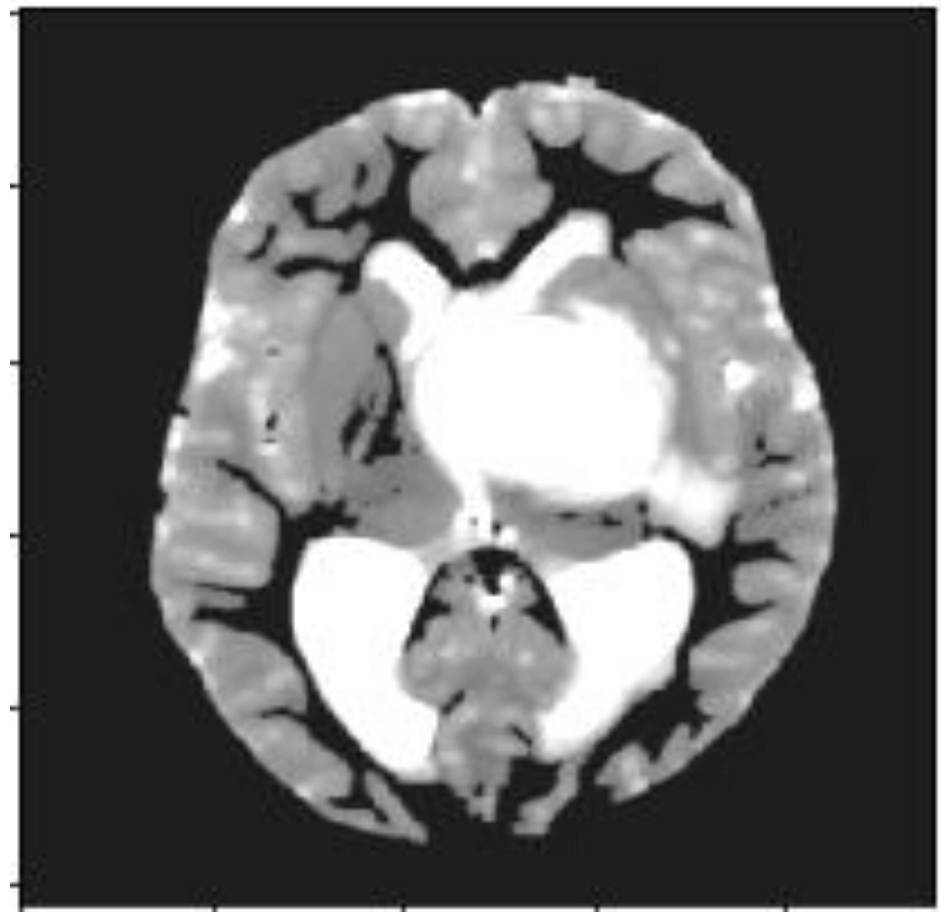


Fig 13.2(a): enhanced MRI

Fig 13.2(b): segmented tumor

Fig 13: segmentation using watershed algorithm

Segmentation Using Thresholding

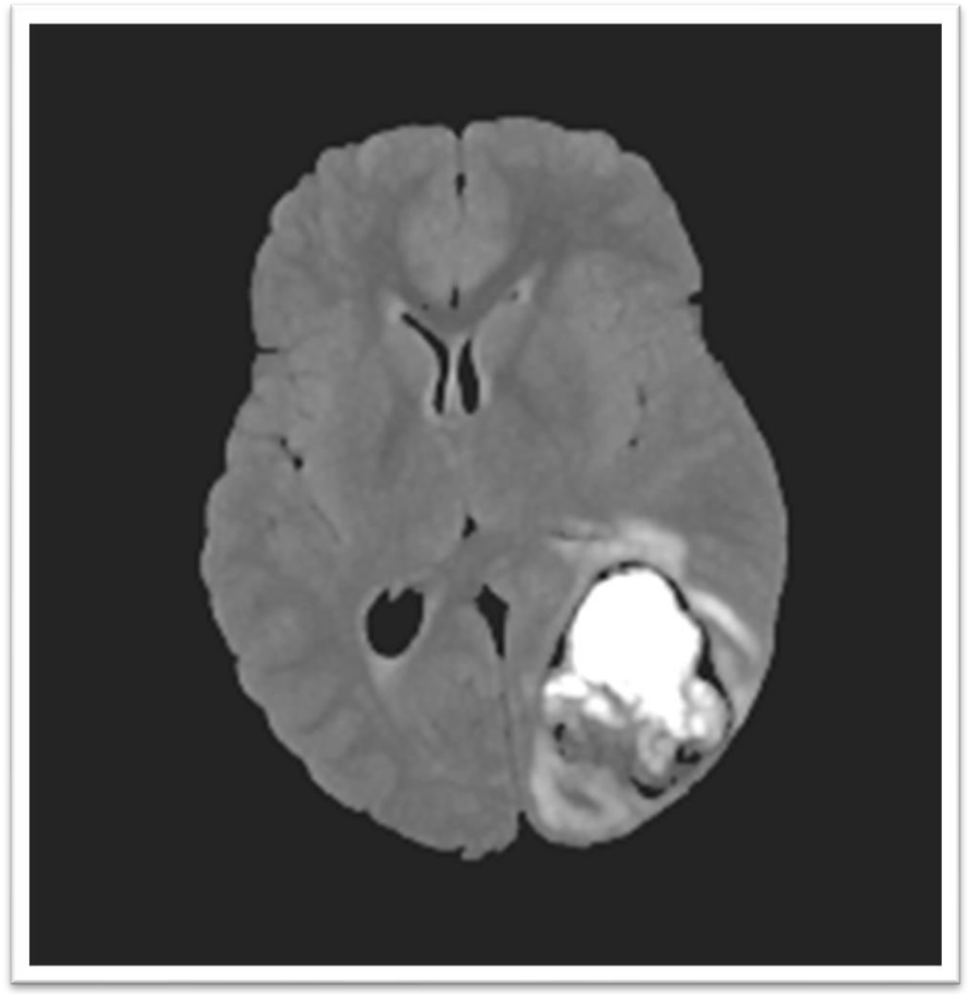


Fig 14.1(a): enhanced MRI

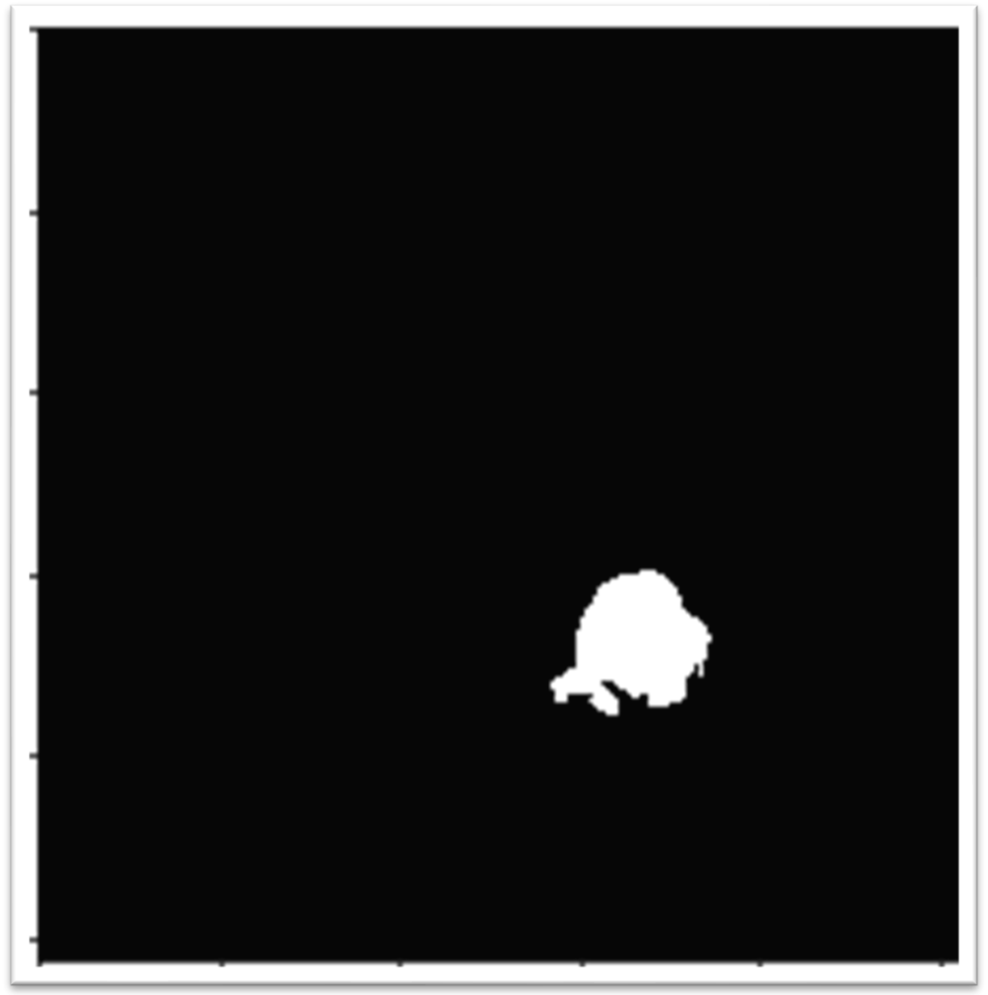


Fig 14.1(b): segmented tumor

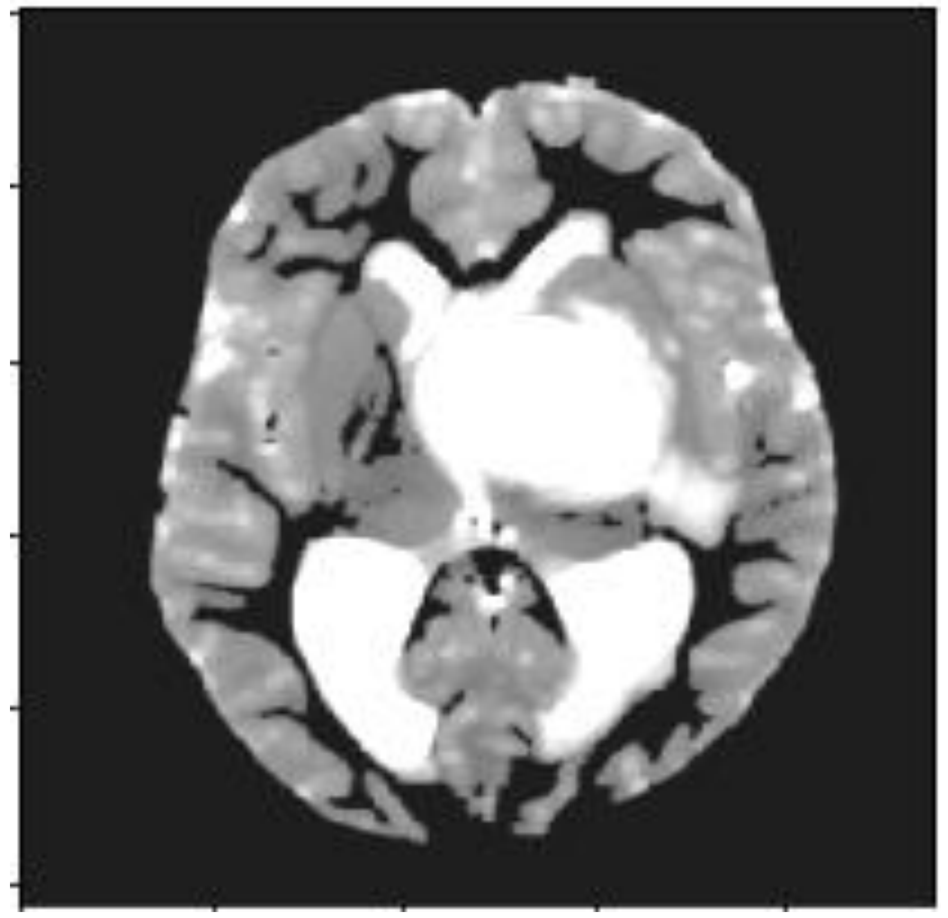


Fig 14.2(a): enhanced MRI

Fig 14.2(b): segmented tumor

Fig 14: segmentation using Thresholding

Segmentation Using Normalize Cut Algorithm

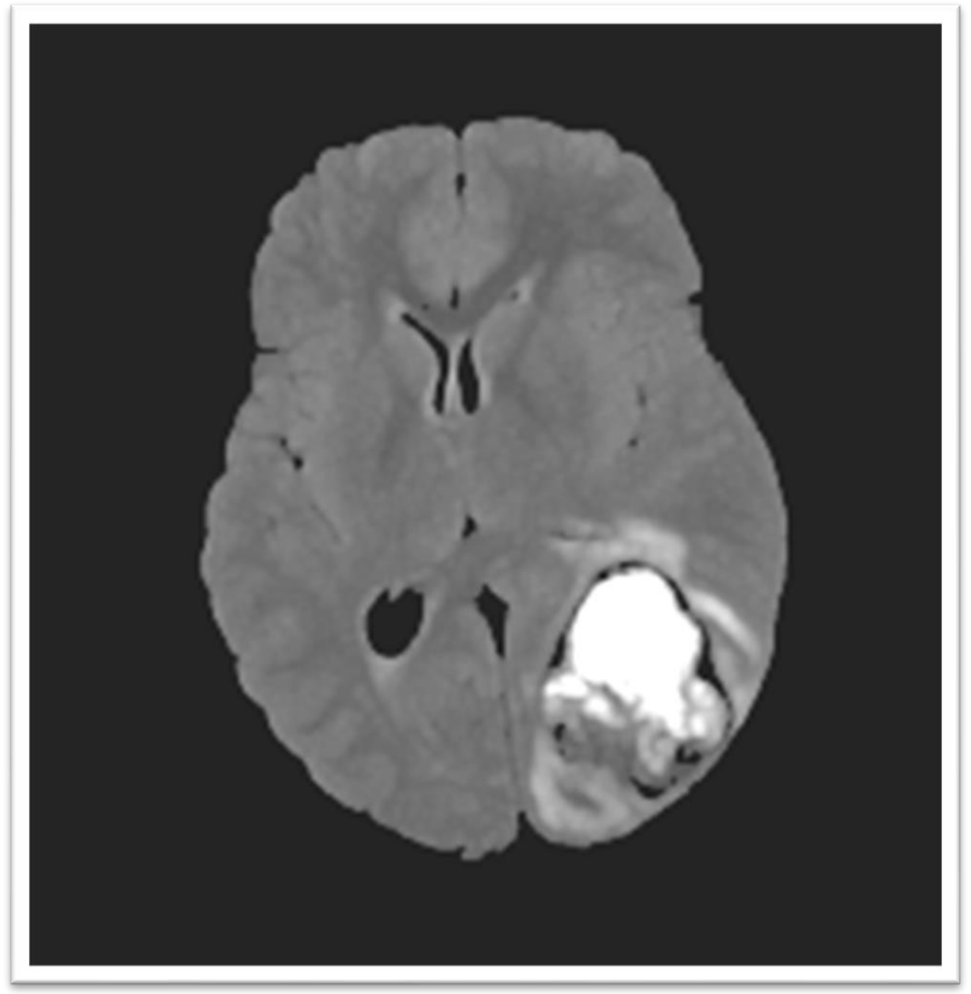


Fig 15.1(a): enhanced MRI

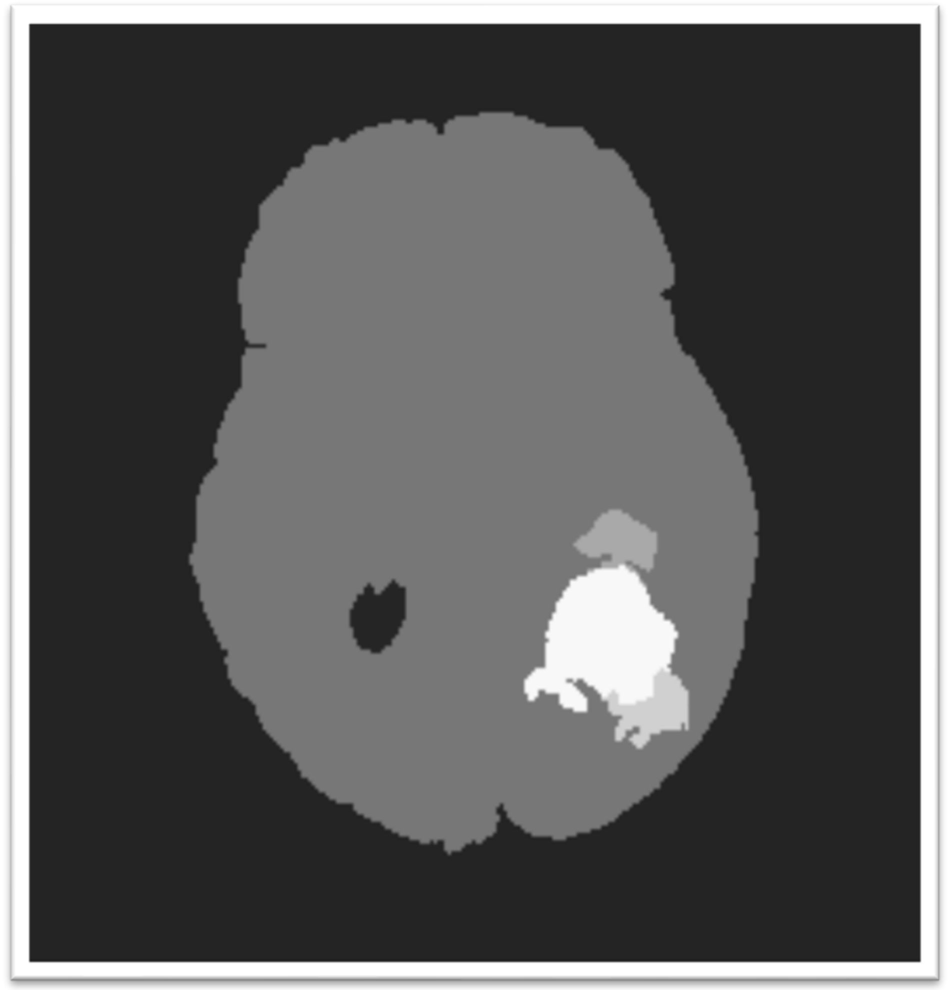


Fig 15.1(b): segmented tumor

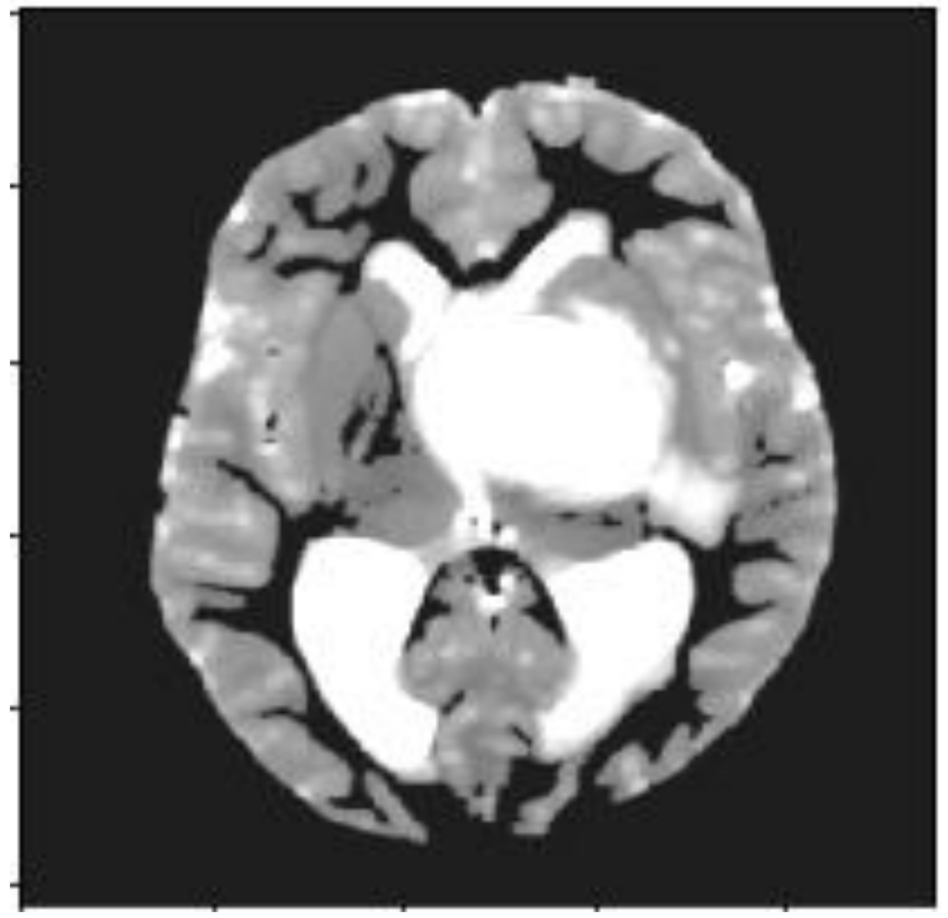


Fig 15.2(a): enhanced MRI

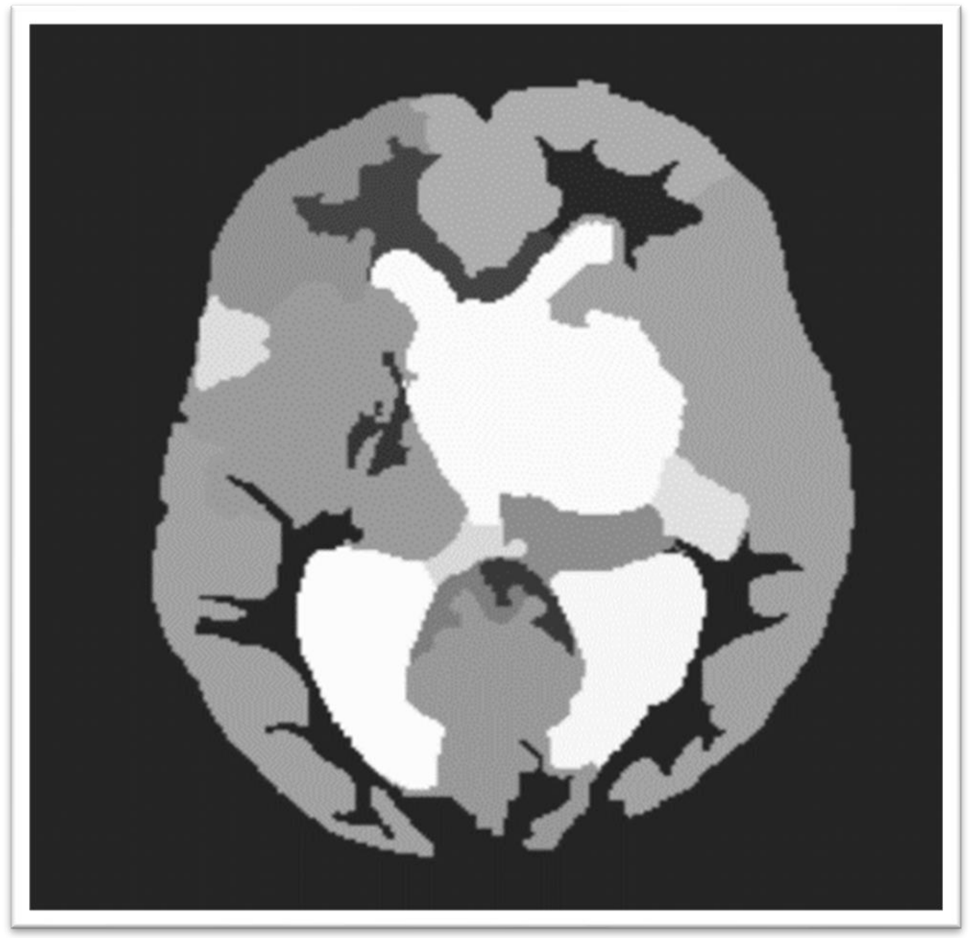


Fig 15.2(b): segmented tumor

Fig 15: segmentation using Normalize Cut Algorithm

Segmentation Comparison

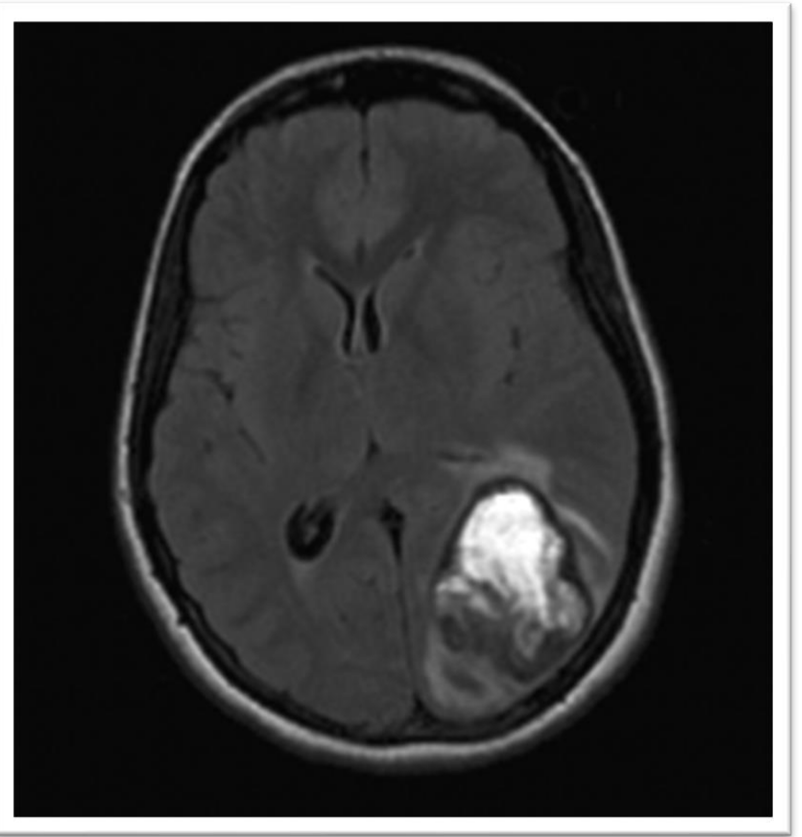


Fig 16.1(a): Input Image

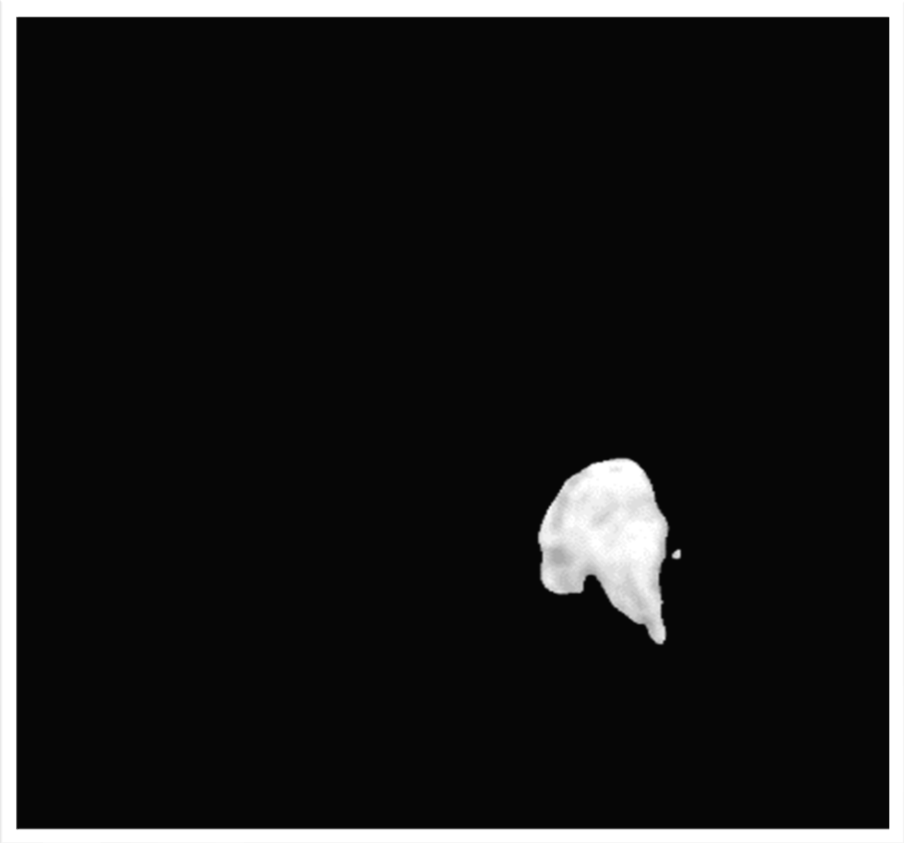


Fig 16.1(b): FCM

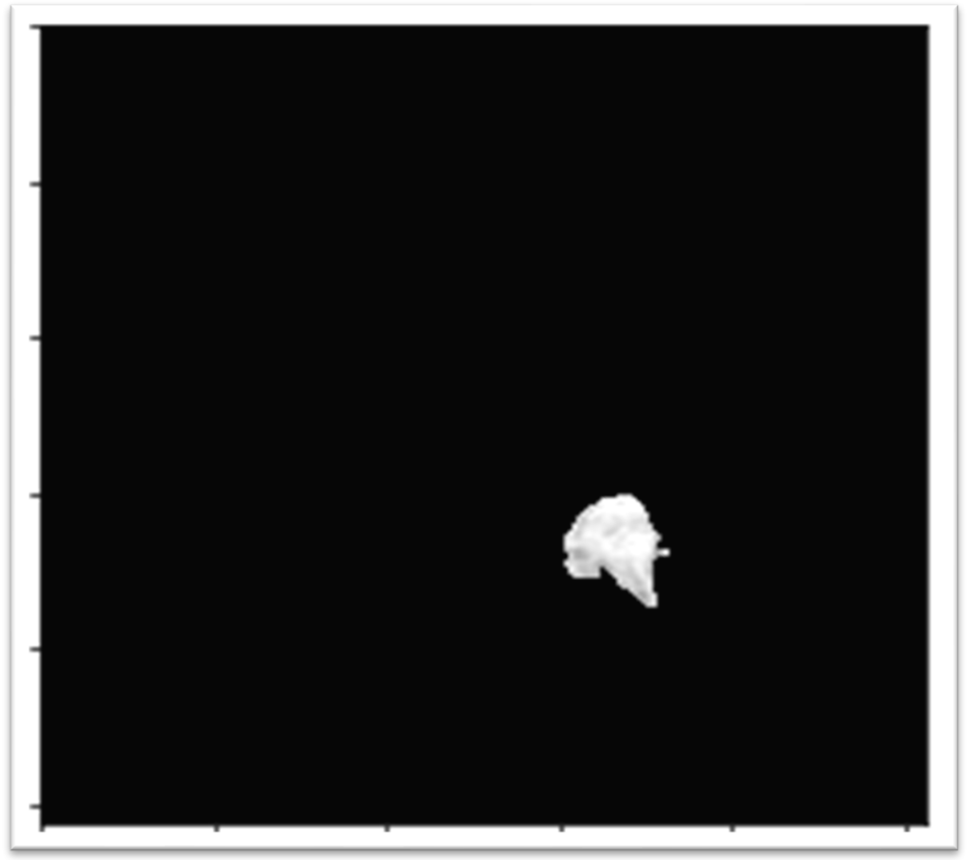


Fig 16.1(c): K-Means

Fig 16.1(d): Thresholding

Fig16.1(d): Normalize Cut

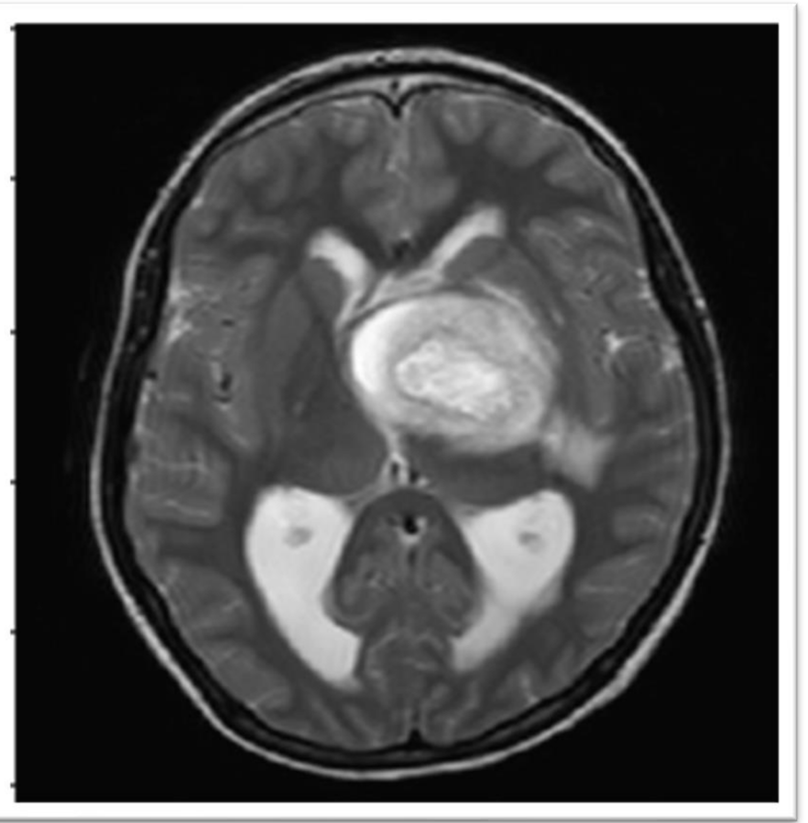


Fig 16.1(a): Input Image

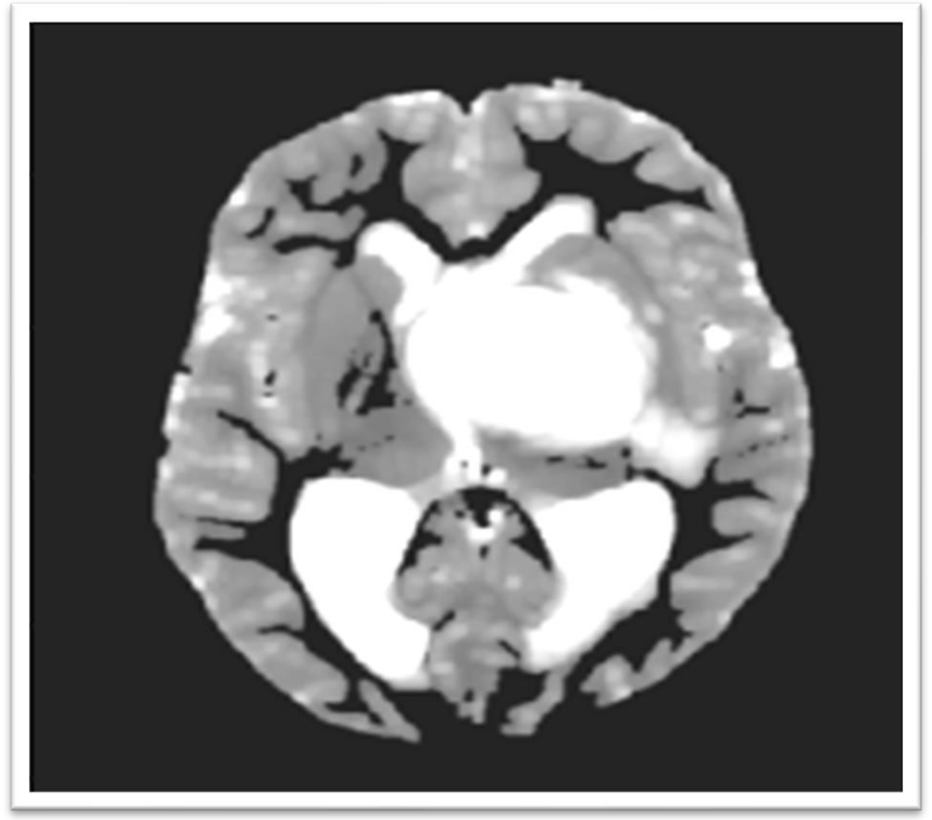


Fig 16.2(a): FCM

Fig 16.2(b): K-Means

Fig 16.2(c): Thresholding

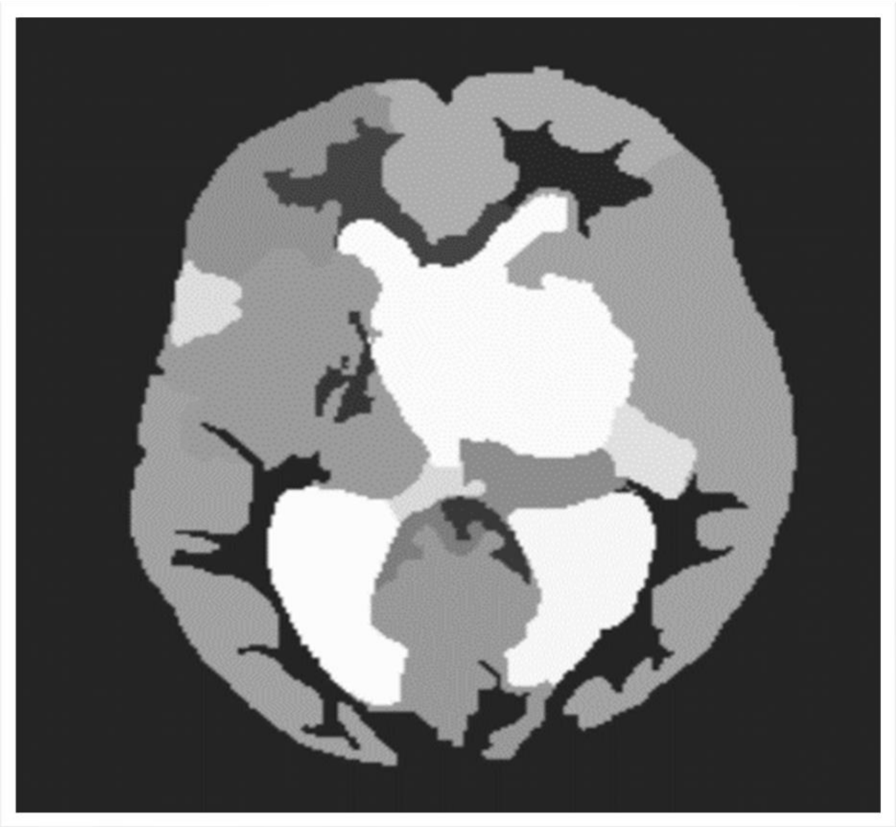
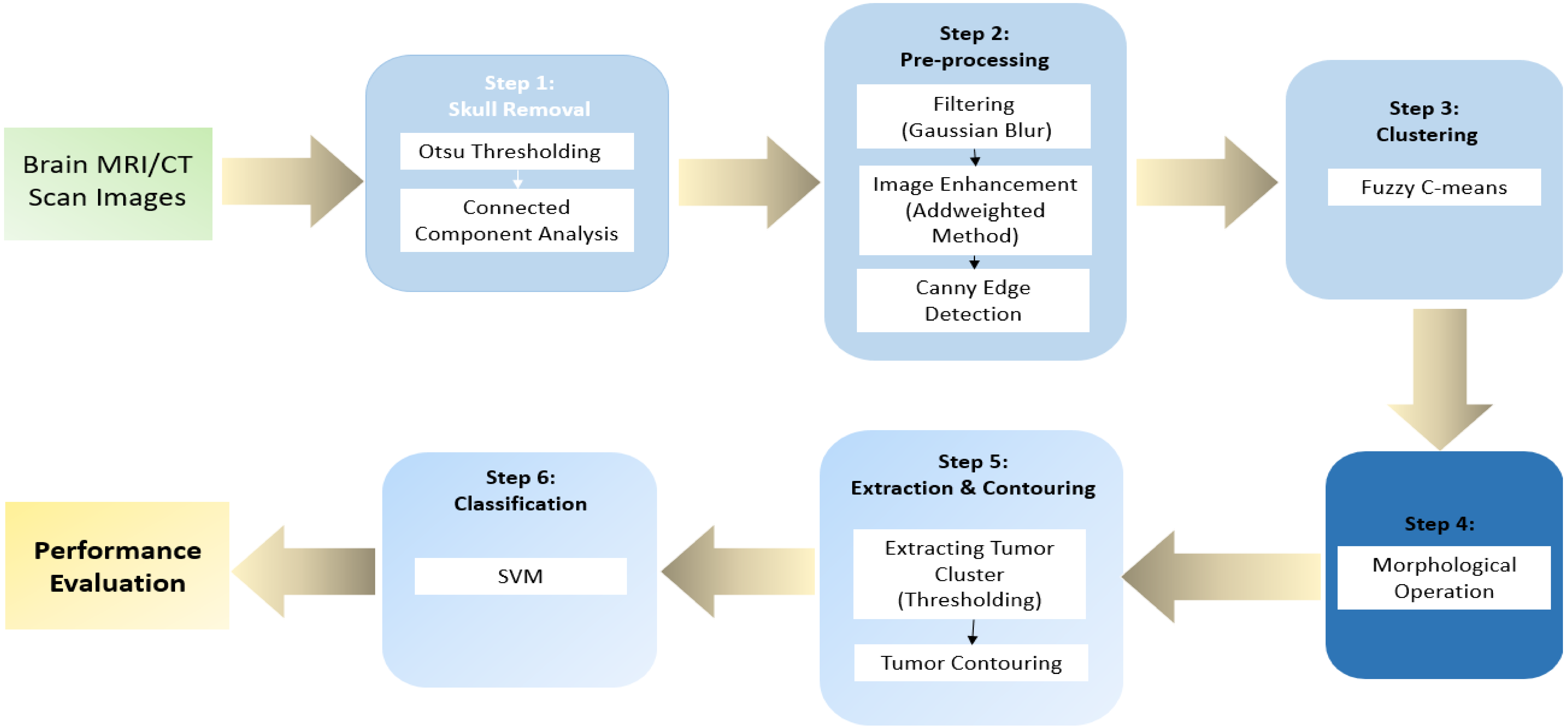


Fig 16.2(d): Normalize Cut

Fig 16: segmentation Comparison

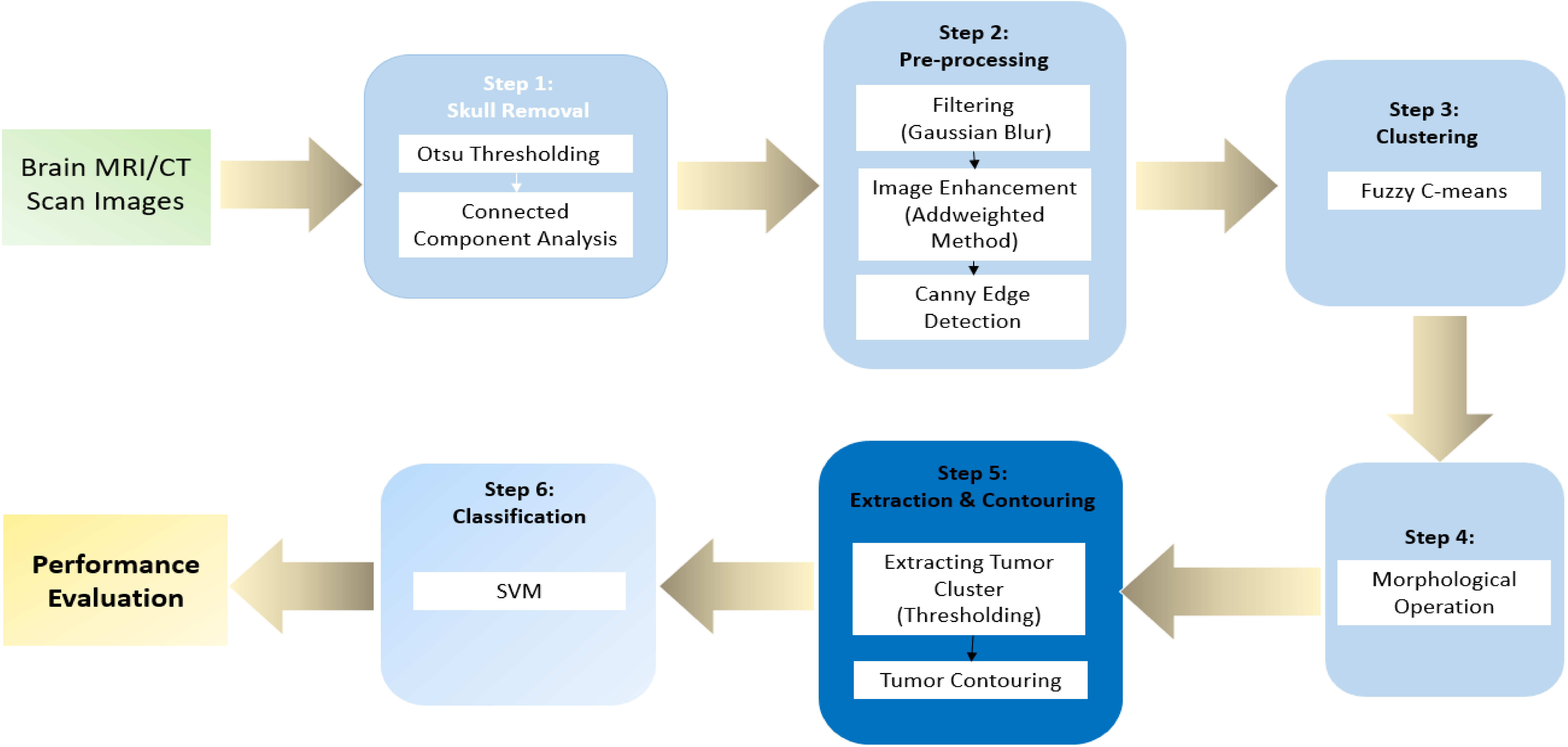
Stage-5: Morphological Operation



Stage-4: Morphological Operation

- Only need the brain part rather than the brain part
- Erosion was done to separate weakly connected regions
- Dilation is applied afterward

Stage-5: Tumor Contouring



Stage-5: Tumor Contouring

- Contours can be explained simply as a curve joining all the continuous points (along the boundary), having same color or intensity
- find the edge of the abnormal tissues by which we can mark the perimeter.
- Used the cv2.findContours( ) method for finding the contours

Stage-5: Tumor Contouring

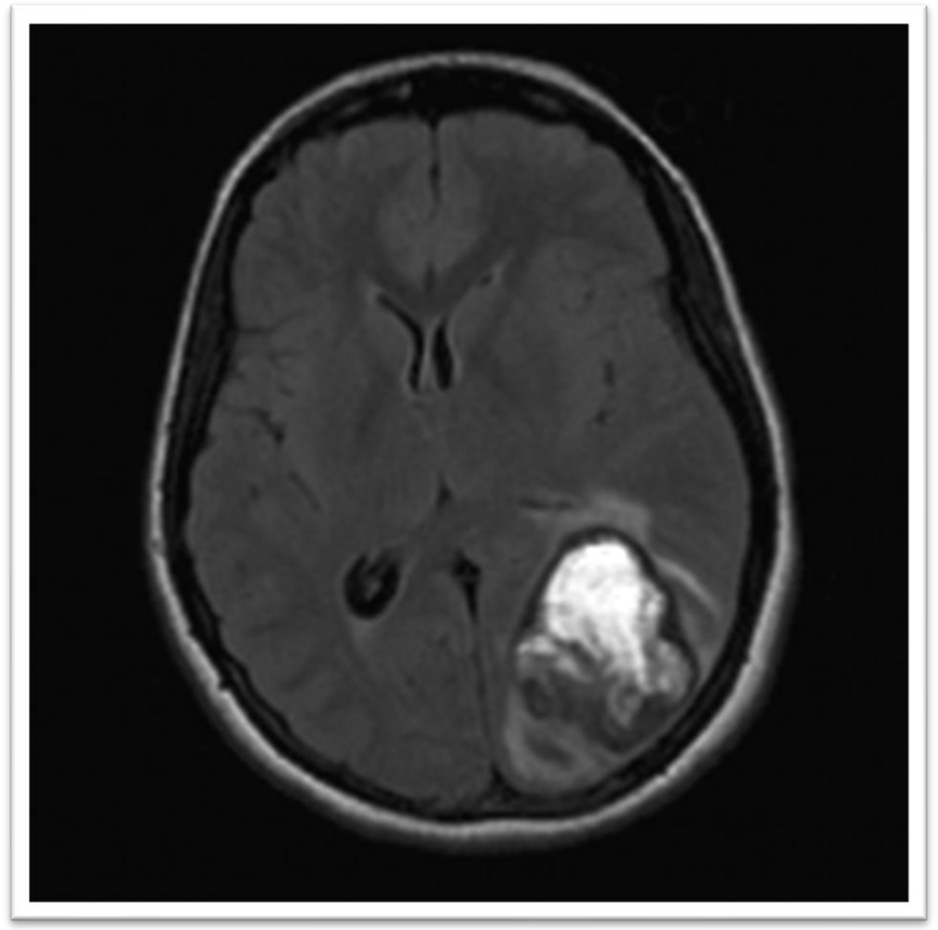


Fig 17.1(a): MRI Image

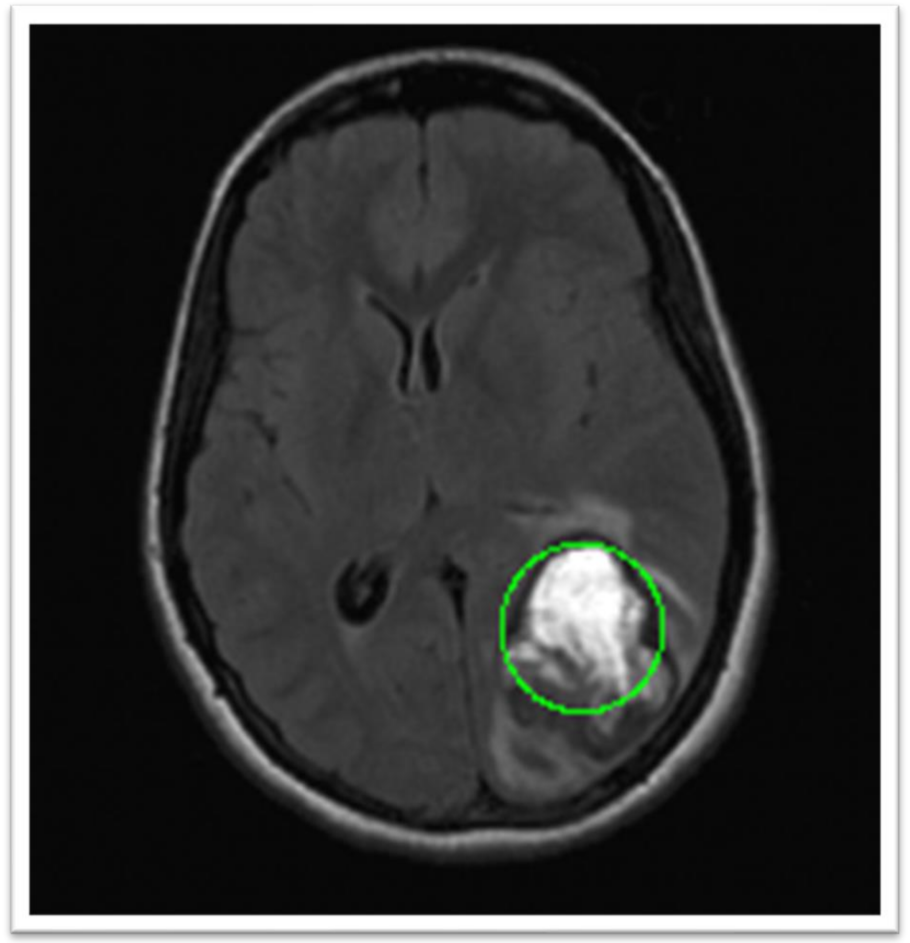


Fig 17.2(b): contoured tumor

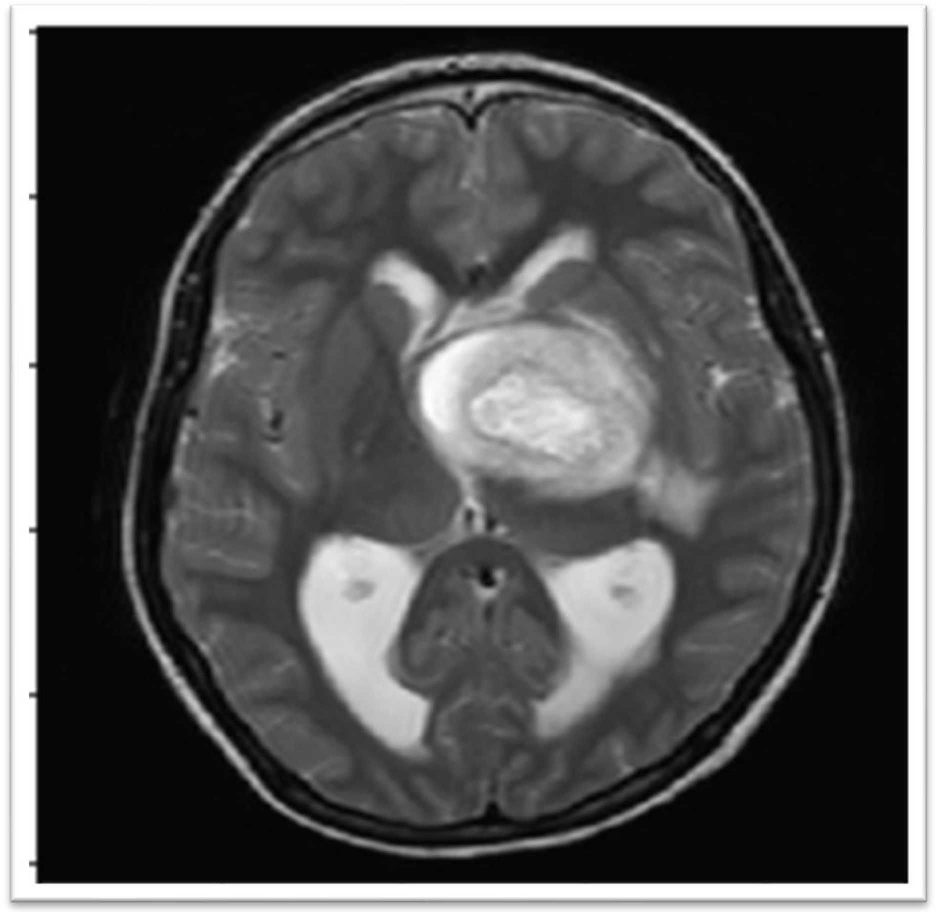


Fig 17.1(a): MRI Image

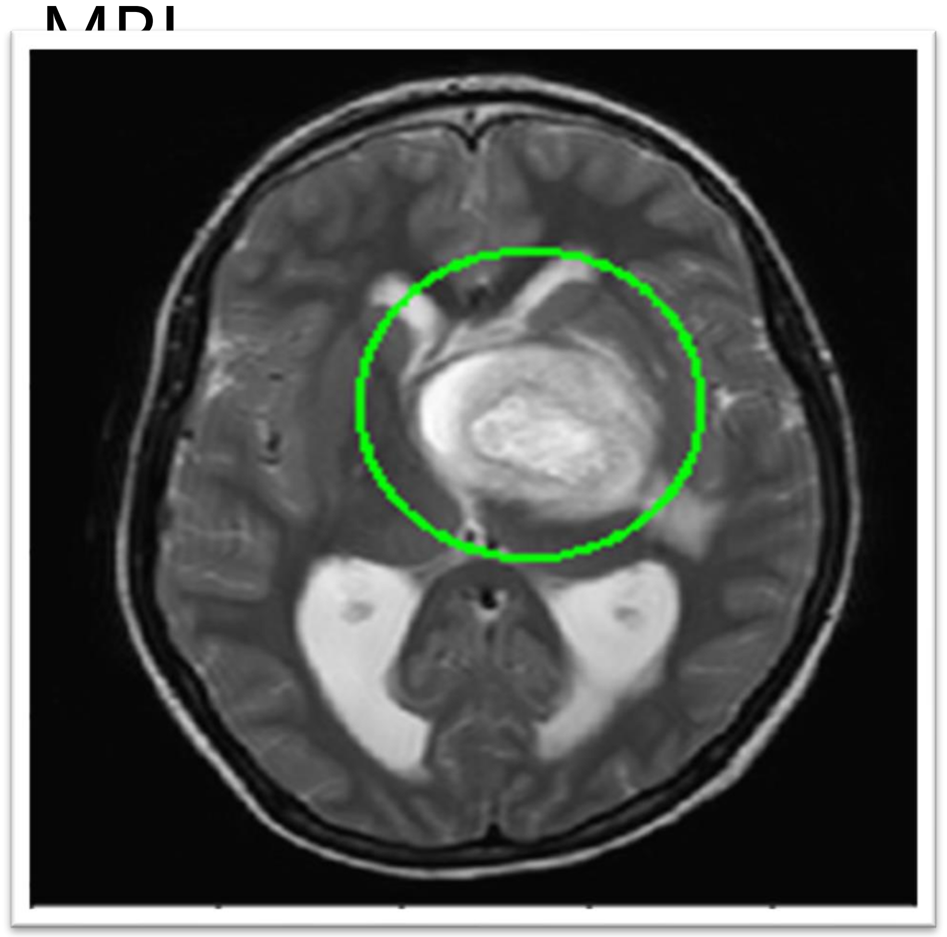


Fig 17.2(b): contoured tumor MRI

Fig 17: tumor contouring

Stage-6: Traditional Classifier

- We adopted six traditional Classifier
  - K-Nearest Neighbor
  - Logistic Regression
  - Multilayer Perceptron
  - Naïve Bayes
  - Random Forest
  - Support Vector Machine
- The model is trained based on two type of splitting ratio
  - Type-1: 80:20 splitting ratio
  - Type-2: 70:30 splitting ratio

METHODOLOGY (CNN)

A Five-Layer CNN developed for tumor detection

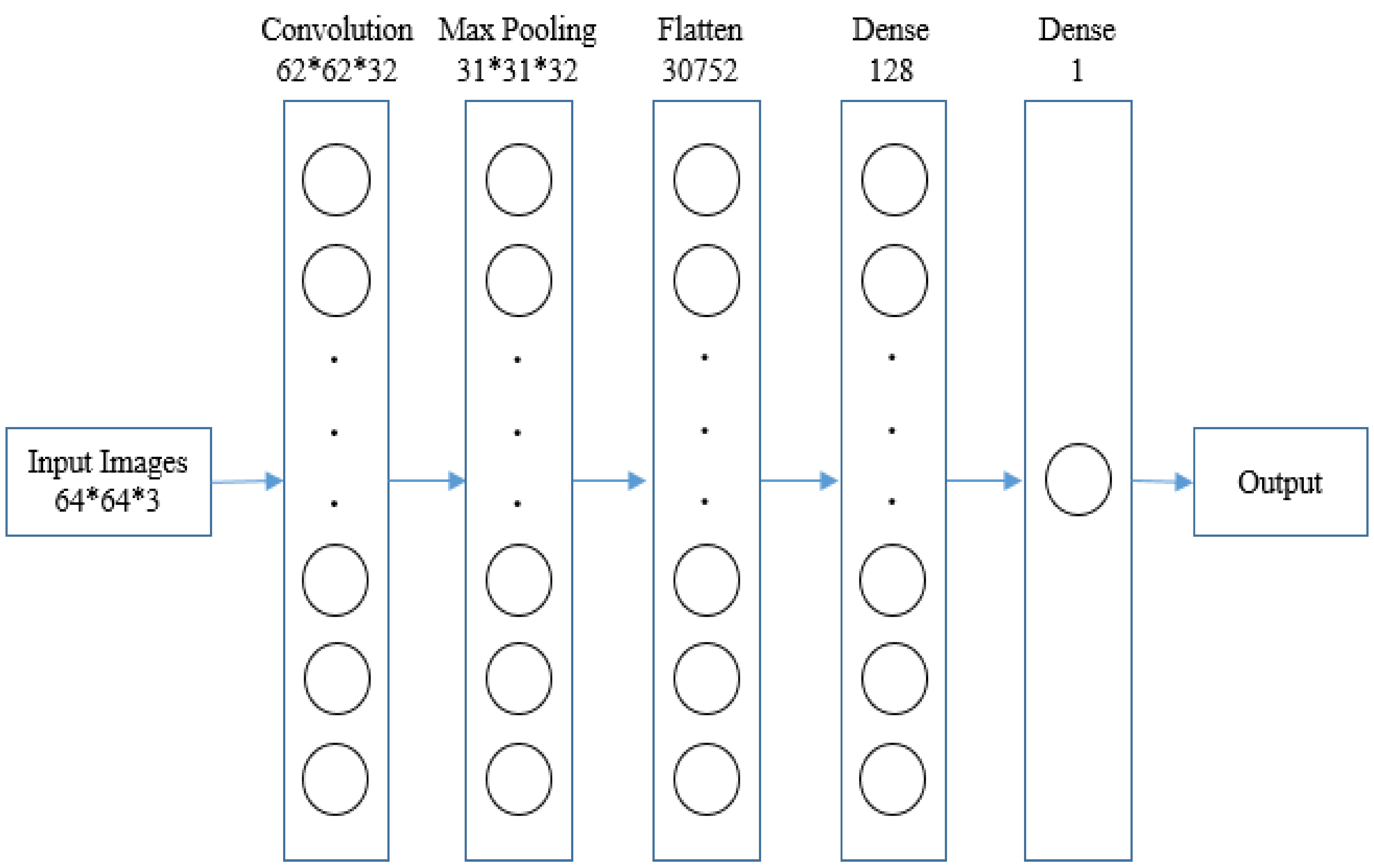


Fig 18: Proposed Methodology for tumor detection using 5-Layer Convolutional Neural Network

Convolution Layer

- The Beginning Layer
- Converting all the images into 64*64*3 homogeneous dimension
- Convolutional kernel of 32 convolutional filters of size 3*3 with the support of 3 tensor channels
- Activation function: ReLU

Max Pooling Layer

- Because of overfitting Max Pooling layer was introduced
- MaxPooling2D for the model
- Runs on 31*31*32 dimension
- Pool size is (2, 2)
- Output: Pooled feature map

Flatten

- Pooled feature map is work as the input
- Transformed the whole matrix into a single column vector
- Fed to the neural network for processing

Fully Connected Layers

- Two fully connected layers were employed Dense-1 and Dense-2 represented the dense layer
- The single obtained vector goes as an input
- Dense function was applied in Keras
- 128 nodes in the hidden layer
- For better Convergence ReLU and sigmoid function is used as an Activation function in the 1^{st} and 2^{nd} dense layer respectively

Workflow of the Model

- **Complete workflow is divided into 7 steps**

Working Flow Devised for Proposed Methodology

1. Load the input dataset
2. Adding a Convolution Layer with 32 convolutional filter
3. Passing the Convolutional kernel into the Max Pooling layer
4. Pooled feature map is used to get the single column vector
5. Processing of the vector in dense layer with 128 nodes
6. Final dense layer applying Sigmoid as the Activation function
7. Validation stage and Performance evaluation

Fig 19: working flow of the proposed CNN Model.

Hyper-parameter values

- The hyper-parameters are divided into two stages- initialization and training

Stage	Hyper-parameter	Value
Initialization	bias	Zeros
	Weights	glorot_uniform
Training	Learning rate	0.001
	beta_1	0.9
	beta_2	0.999
	epsilon	None
	decay	0.0
	amsgrad	False
	epoch	10
	Batch_size	32
	steps_per_epoch	80

Table I: HYPER-PARAMETER VALUE OF CNN MODEL

Evaluation Process

- **We devised an algorithm for the performance evaluation of our proposed model**

Algorithm 1: Evaluation process of CNN model

```
loadImage();
dataAugmentation();
splitData();
loadModel();
for each epoch in epochNumber do
    for each batch in batchSize do
        ŷ = model(features);
        loss = crossEntropy(y, ŷ);
        optimization(loss);
        accuracy();
        bestAccuracy = max(bestAccuracy, accuracy);
return
```

Fig 20: algorithm of the performance evaluation

Experimental Result

I – Traditional Machine Learning

Type-1: 70:30 splitting ratio

Classifiers	Accuracy (%)	Recall	Specificity	Precision	Dice Score	Jaccard Index
K-Nearest Neighbor	89.39	0.949	0.428	0.933	0. 941	0.889
Logistic Regression	87.88	0.949	0.286	0.918	0.933	0.875
Multilayer Perception	89.39	**1.000**	0.167	0.894	0.944	0.894
Naïve Bayes	78.79	0.797	**0.714**	**0.959**	0.870	0.770
Random Forest	89.39	0.983	0.167	0.903	0.943	0.892
SVM	**92.42**	0.983	0.428	0.935	**0.959**	**0.921**

Table II: confusion matrices of the classifiers

Type-1: 70:30 splitting ratio

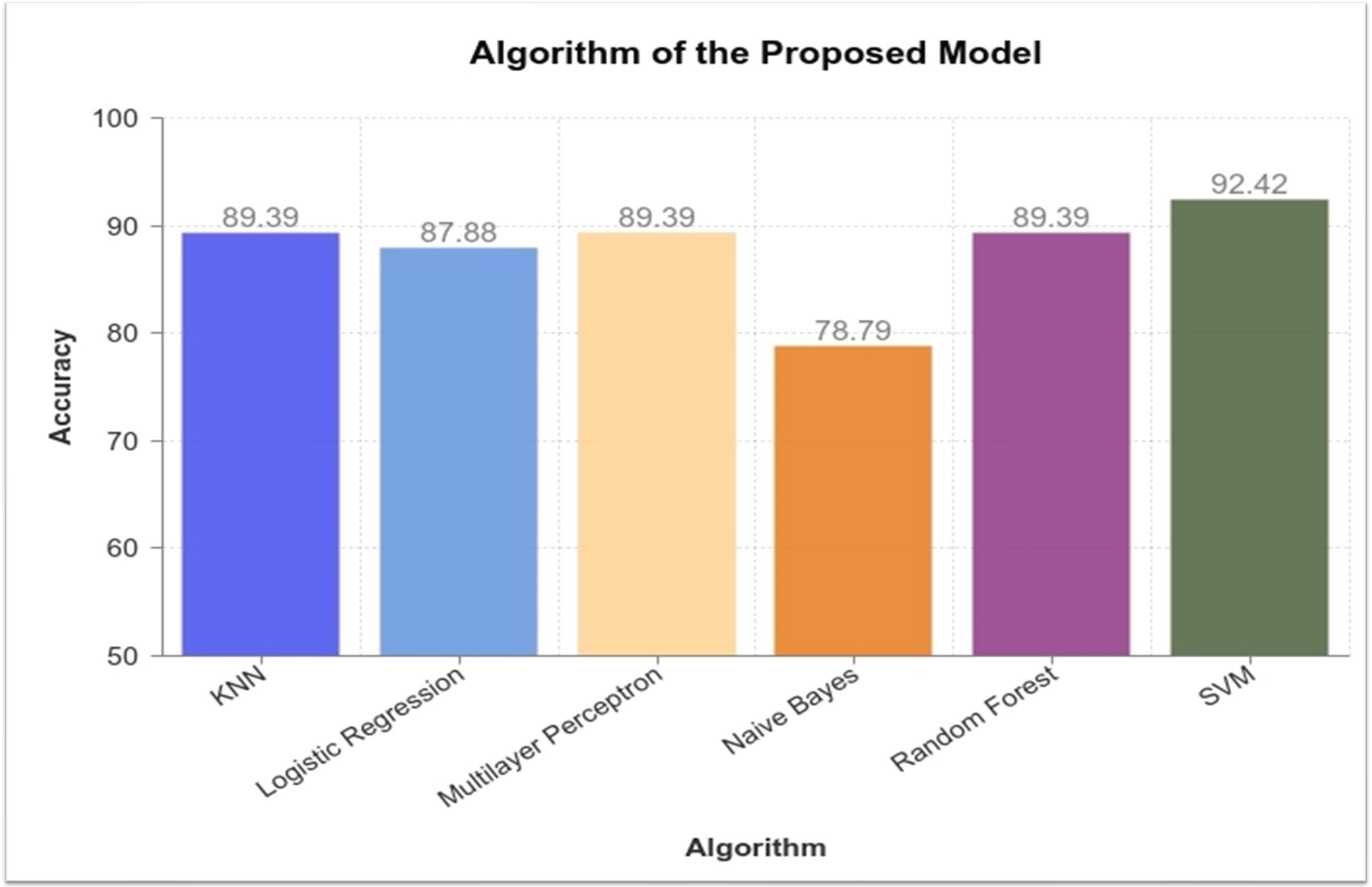


Fig 21: accuracy of the classifiers

Type-2: 80:20 splitting ratio

Classifiers	Accuracy (%)	Recall	Specificity	Precision	Dice Score	Jaccard Index
K-Nearest Neighbor	84.09	0.40	0.8810	**0.48**	**0.465**	0.412
Logistic Regression	**88.63**	0.50	0.9050	0.20	0.285	0.167
Multilayer Perception	**88.63**	0.41	0.8864	**0.48**	0.442	**0.465**
Naïve Bayes	77.27	0.22	**0.9143**	0.40	0.285	0.167
Random Forest	**88.63**	0.50	0.9050	0.20	0.285	0.167
SVM	**88.63**	0.50	0.9050	0.20	0.285	0.167

Table III: confusion matrices of the classifiers

Type-2: 80:30 splitting ratio

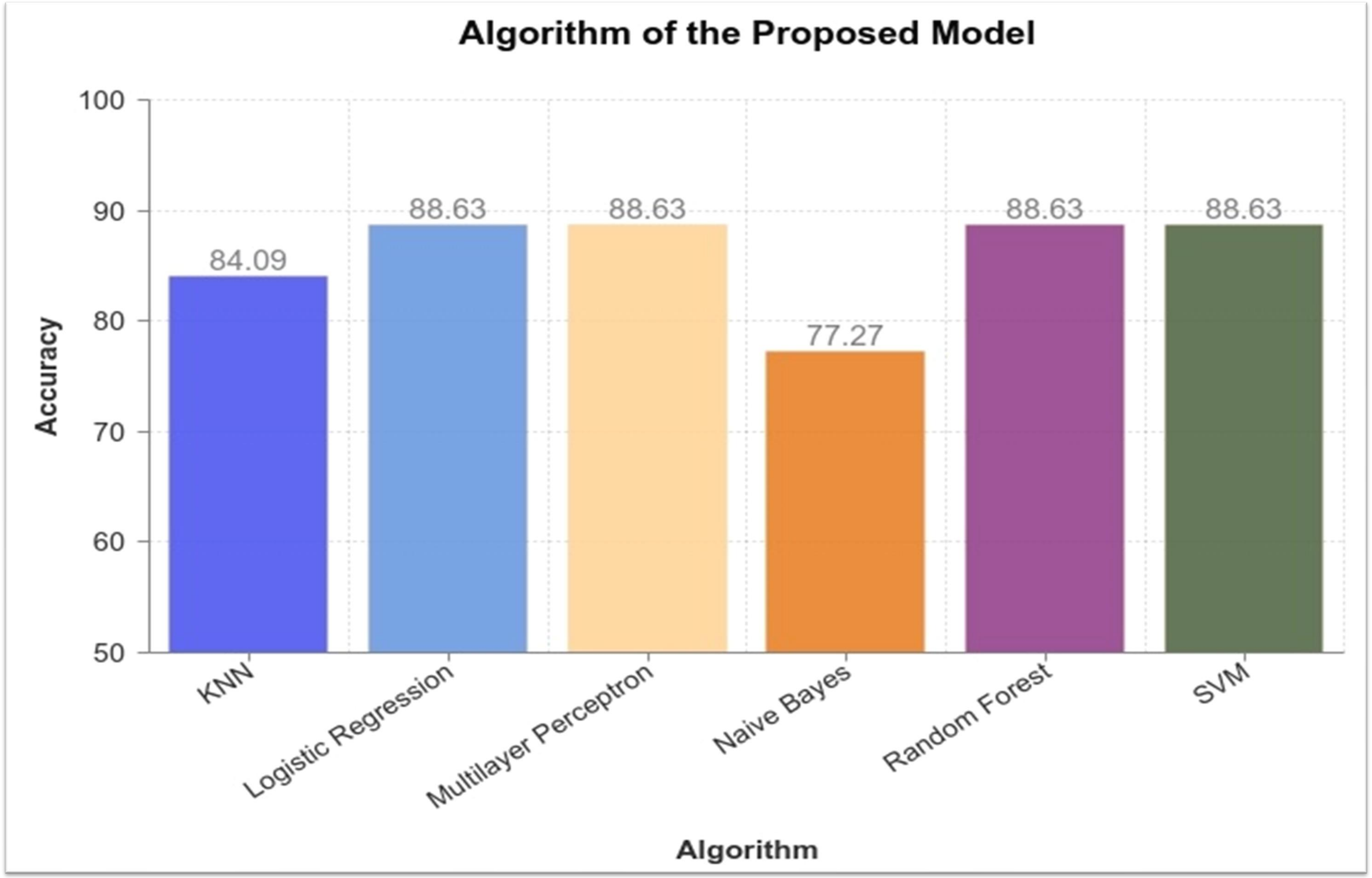


Fig 22: accuracy of the classifiers

II – CNN

Experiment-I

Learning Rate	Epochs	Time to train(sec)	Accuracy (%)
0.001	10	180	92.88
	20	231	93.23
	50	**500**	**96.55**
	100	1227	96.01
0.005	10	198	93.00
	20	**240**	**93.13**
	50	555	90.68
	100	1133	91.22
0.01	10	191	88.76
	20	235	90.92
	50	634	91.25
	100	**1347**	**93.45**

Fig 23: Training Time and Accuracy of the proposed CNN model (splitting ratio 70:30)

Experiment-II

Learning Rate	Epochs	Time to train(sec)	Accuracy(%)
0.001	**10**	**175**	**97.87**
	20	**233**	**97.87**
	50	527	95.74
	100	1200	95.69
0.005	10	177	96.03
	20	**203**	**97.62**
	50	488	95.55
	100	1027	95.55
0.01	10	178	92.09
	20	200	93.04
	50	599	**93.77**
	100	966	92.00

Fig 24: Training Time and Accuracy of the proposed CNN model (splitting ratio 80:20)

Experiment-III

Split Ratio	Batch Size	Epoch	CNN Hidden Layer	Max Pooling	Accuracy(%)
80:20	32	8	62*62*32	31*31*32	92.72
		9	62*62*32	31*31*32	85.82
		10	62*62*32	31*31*32	86.85
		11	**62*62*32**	**31*31*32**	**87.88**
	64	8	62*62*32	31*31*32	93.67
		9	62*62*32	31*31*32	94.98
		10	**62*62*32**	**31*31*32**	**95.74**
		11	**62*62*32**	**31*31*32**	**95.74**
		12	62*62*32	31*31*32	94.89
		10	62*62*32	31*31*32	95.72
			62*62*32		
			62*62*32	31*31*32	91.67
			62*62*32		
			31*31*128		
			62*62*32	31*31*32	91.99
			31*31*128	15*15*128	

Fig 25: Accuracy of the proposed model based on batch size (splitting ratio 80:20)

Experiment-IV

Split Ratio	Batch Size	Epoch	CNN Hidden Layer	Max Pooling	Accuracy(%)
70:30	32	8	62*62*32	31*31*32	81.35
		9	62*62*32	31*31*32	83.71
		10	62*62*32	31*31*32	87.87
		11	**62*62*32**	**31*31*32**	**89.13**
	64	8	62*62*32	31*31*32	88.07
		9	62*62*32	31*31*32	88.76
		10	62*62*32	31*31*32	91.23
		11	**62*62*32**	**31*31*32**	**94.90**
		12	62*62*32	31*31*32	94.81
		10	62*62*32	31*31*32	93.72
			62*62*32		
			62*62*32	31*31*32	91.82
			62*62*32		
			31*31*128		
			62*62*32	31*31*32	91.07
			31*31*128	15*15*128	

Fig 26: Accuracy of the proposed model based on batch size (splitting ratio 70:30)

Experiment-V

<table>
<tr><th>Split Ratio</th><th>Batch Size</th><th>Epoch</th><th>CNN Hidden Layer</th><th>Max Pooling</th><th>Accuracy(%)</th></tr>
<tr><td rowspan="16">60:40</td><td rowspan="4">32</td><td>8</td><td>62*62*32</td><td>31*31*32</td><td>76.10</td></tr>
<tr><td>9</td><td>62*62*32</td><td>31*31*32</td><td>79.66</td></tr>
<tr><td>10</td><td>62*62*32</td><td>31*31*32</td><td>83.82</td></tr>
<tr><td>11</td><td>62*62*32</td><td>31*31*32</td><td>85.17</td></tr>
<tr><td rowspan="12">64</td><td>8</td><td>62*62*32</td><td>31*31*32</td><td>83.89</td></tr>
<tr><td>9</td><td>62*62*32</td><td>31*31*32</td><td>86.23</td></tr>
<tr><td>10</td><td>62*62*32</td><td>31*31*32</td><td>90.09</td></tr>
<tr><td>11</td><td>62*62*32</td><td>31*31*32</td><td>90.27</td></tr>
<tr><td>12</td><td>62*62*32</td><td>31*31*32</td><td>92.13</td></tr>
<tr><td rowspan="7">10</td><td>62*62*32</td><td rowspan="2">31*31*32</td><td rowspan="2">90.63</td></tr>
<tr><td>62*62*32</td></tr>
<tr><td>62*62*32</td><td rowspan="3">31*31*32</td><td rowspan="3">89.62</td></tr>
<tr><td>62*62*32</td></tr>
<tr><td>31*31*128</td></tr>
<tr><td>62*62*32</td><td>31*31*32</td><td rowspan="2">93.36</td></tr>
<tr><td>31*31*128</td><td>15*15*128</td></tr>
</table>

Fig 27: Accuracy of the proposed model based on batch size (splitting ratio 60:40)

Model Accuracy Curve

- The curve represents training and validation accuracy of the model
- Underfitted for 1 time, overfitted for 2 times
- Prediction value in the last of the model is high, so the accuracy is high

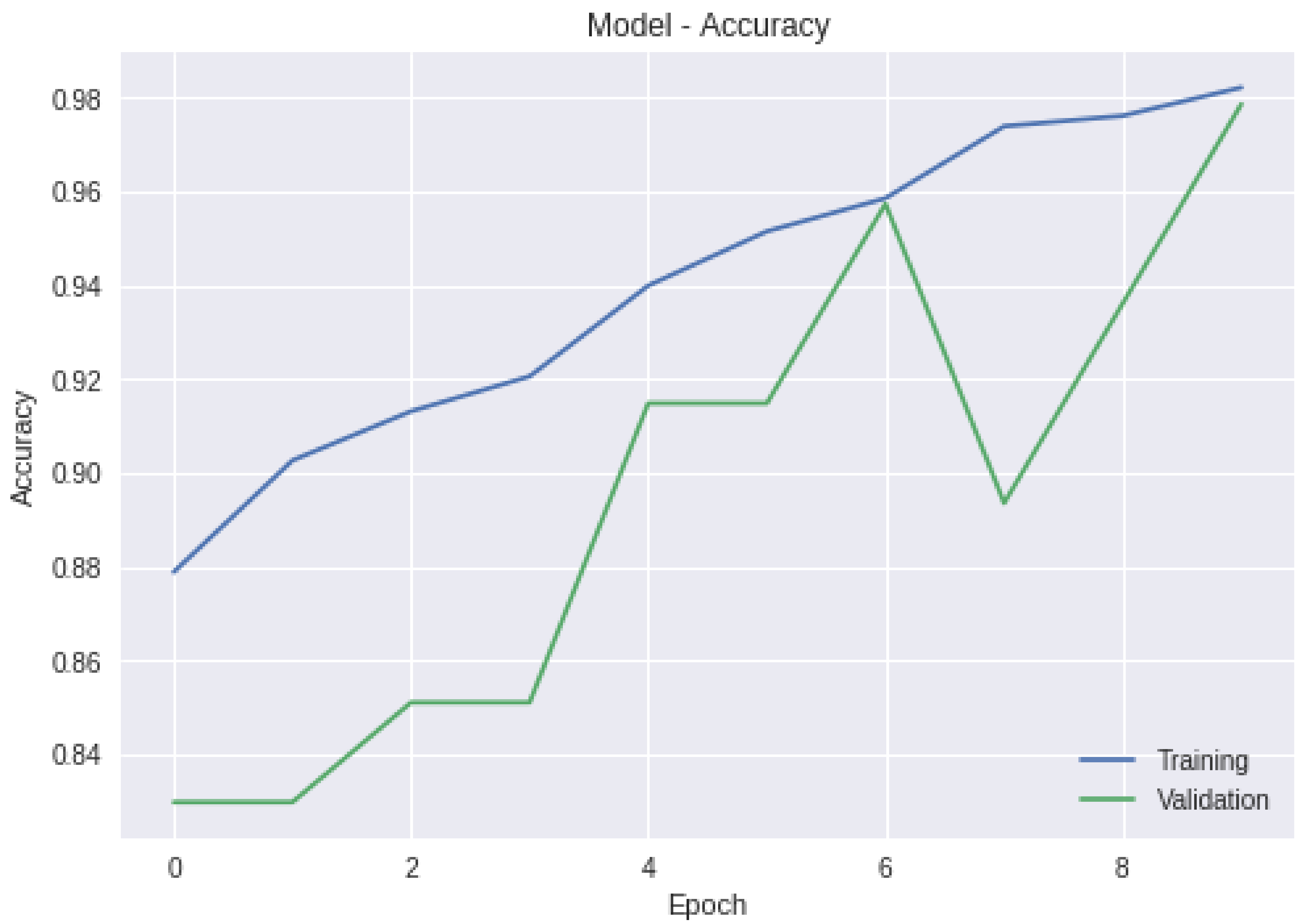


Fig 27: Accuracy Curve of the proposed CNN model (splitting ratio - 80:20)

Model Loss Curve

- The actual loss per epoch represents the graph
- Estimate the loss of the model
- Initially no prediction so the loss function is high and up to 10 epochs it is gradually decreased.

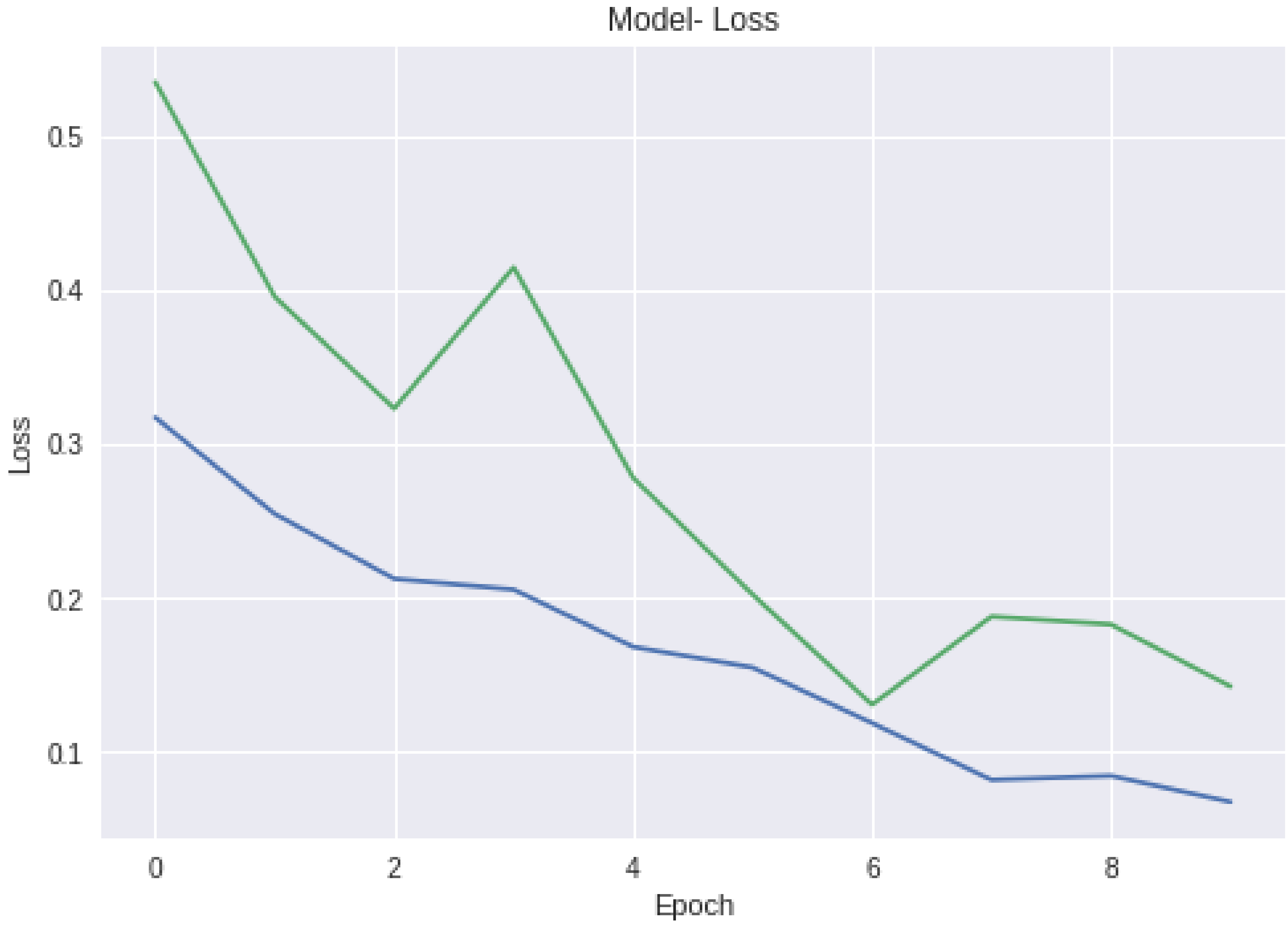


Fig 28: Loss Curve of the proposed CNN model (splitting ratio - 80:20)

Learning Rate vs Training Time

- ✓ Learning Rate 0.01 is the best performer for the best output

Fig 29: Learning rate vs training time curve

Learning Rate vs Accuracy

- More accurate result we can find in less learning rate in comparison.

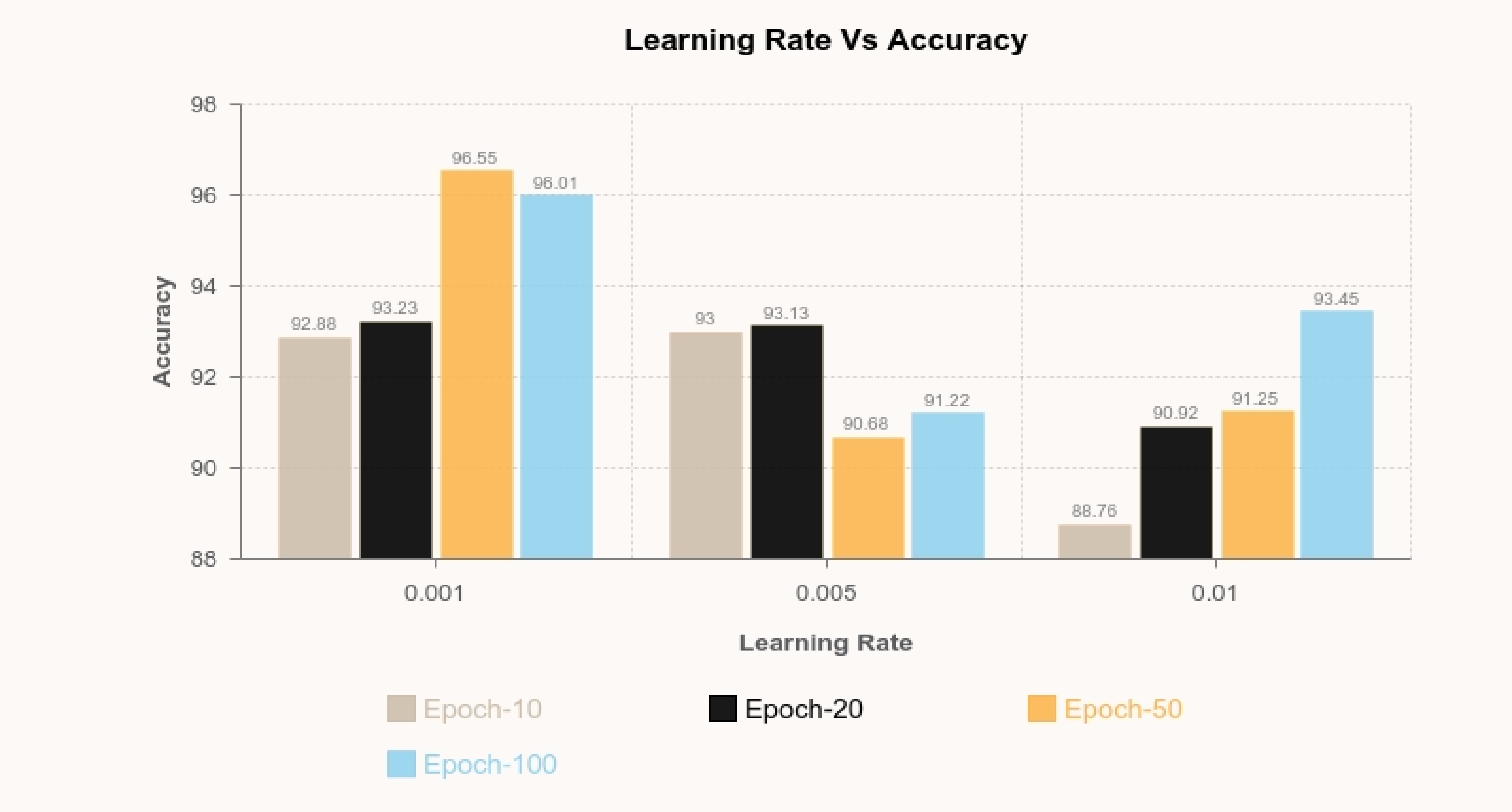


Fig 30: Learning rate vs Accuracy curve

Performance Comparison(1)

- The best performance we have gotten from the dataset in CNN is **97.87%**

Model	Splitting Ratio	Accuracy (%)
Traditional Machine Learning	70:30	92.42
	80:20	88.63
CNN	60:40	93.36
	70:30	96.55
	80:20	**97.87**

Fig 31: Performance comparison of the proposed traditional machine learning and CNN model

Performance Comparison(2)

No	Paper Name	Year	Method	Accuracy
1	Brain tumor segmentation based on a new threshold approach[2]	2017	Pixel Subtraction + Thresholding	96%
2	Computer Aided System for Brain Tumor Detection and Segmentation [35]	2011	Morphological Operation + Windowing	97%
3	Brain tumor detection and segmentation using conditional random field [48]	2015	Conditional Random Field	79%
4	Detection of Brain Tumor in MRI Images, using Combination of Fuzzy C-Means and SVM [49]	2015	FCM + SVM	91.66%
5	Image Analysis for MRI Based Brain Tumor Detection and Feature Extraction Using Biologically Inspired BWT and SVM [71]	2017	Berkeley Wavelet Transform + SVM	96.51%
6	Fuzzy Clustering and Deformable Model for Tumor Segmentation on MRI Brain Image: A Combined Approach [74]	2011	FCM + Deformable Model	82.1% (Jaccard Index)
7	A Segmentation based Automated System for Brain Tumor Detection [101]	2016	Morphological Operation + Thresholding	84.72%
8	**Proposed Model**	**2019**	**FCM + Five Layer CNN Model**	**97.87%**

Fig 32: Performance Comparison with the existing works

Limitations

Limitations

- Well adaptation of automated medical image analysis in the perspective of Bangladesh
- Early detection of Brain Tumors
- Reducing the pressure on Human judgement
- Build a User Interface which can identify the cancerous cells
- Reducing the death rate by early detection
- Supporting faster communication, where patient care can be extended to remote areas

Publications

Publications

- **Brain Tumor Segmentation Techniques on Medical Images - A Review**
  International Journal of Scientific & Engineering Research Volume 10, Issue 2, February-2019, ISSN 2229-5518

- **Brain Tumor Detection Using Convolutional Neural Network**
  Tonmoy Hossain, Fairuz Shadmani Shishir, Mohsena Ashraf, MD Abdullah Al Nasim, Faisal Muhammad Shah

  1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT-2019), May 3-5, 2019, East West University, Dhaka, Bangladesh

FUTURE PLAN

Future Plan

- ✓ Work on 3D images
- ✓ Build our own dataset based on Bangladeshi patients
- ✓ Try to detect the grade and stage of the tumor
- ✓ Try to predict the location of the tumor from 3D images

THANK YOU!

Any Question!