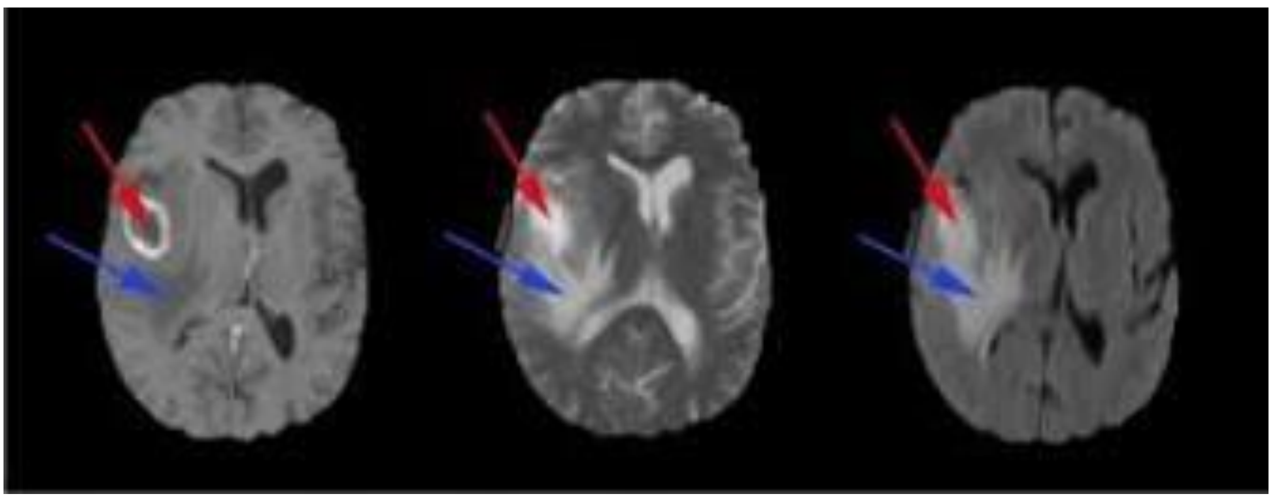
BRAIN TUMOR DETECTION USING ENHANCED SOBEL FILTER



BRAIN TUMOR DETECTION USING ENHANCED SOBEL FILTER

Submitted by:

Reddy jyothi sri D
(RA2011028010108)

1. Abstract

- The early detection of brain tumors is crucial for successful treatment and improved patient outcomes. In recent years, machine learning and computer vision techniques have been utilized for automatic detection and classification of brain tumors in medical imaging. This paper presents a review of the current state-of-the-art techniques for brain tumor detection using various imaging modalities such as magnetic resonance imaging (MRI), computed tomography (CT), and positron emission tomography (PET). The review covers the different approaches used for preprocessing, feature extraction, and classification of brain tumors.
- Furthermore, it highlights the challenges and limitations of current techniques and provides insights into future research directions. Overall, the use of machine learning and computer vision techniques for brain tumor detection has shown promising results and has the potential to improve the accuracy and speed of diagnosis, ultimately leading to better patient outcomes.

2. Objectives

The objectives of brain tumor detection are as follows:

- Early Detection: The primary objective of brain tumor detection is to detect the presence of a tumor at an early stage, when it is still small and treatable. Early detection can help improve the chances of successful treatment and reduce the risk of complications.

- Accurate Diagnosis: Accurate diagnosis is crucial for determining the type, location, and size of the tumor. This information is essential for planning the most effective treatment strategy.

- Efficient Detection: Efficient detection of brain tumors is essential for minimizing the time required for diagnosis and treatment. This can help reduce patient anxiety and improve overall treatment outcomes.

- Non-Invasive Detection: Non-invasive detection techniques, such as magnetic resonance imaging (MRI) and computed tomography (CT), are preferred over invasive techniques, such as biopsies, as they are less risky and more comfortable for the patient.
- Reliable and Reproducible Results: The detection techniques used for brain tumors should provide reliable and reproducible results that can be easily interpreted by medical professionals.

- Overall, the objectives of brain tumor detection are to improve patient outcomes by detecting tumors at an early stage, accurately diagnosing the tumor, and providing efficient and non-invasive detection methods that produce reliable and reproducible results.

3. Literature Survey:

Paper	Author	Abstract	Description
EDGE DETECTION FOR BRAIN TUMOR PATTERN RECOGNITION	Riries Rulaningtyas and Khusnul Ain	Brain tumor diagnosis is done by doctors. For detecting brain tumor grading always gives different conclusion between one doctor to another. For helping doctors diagnose brain tumor grading, this research made a software with edge detections method , so it could give edge pattern of brain and brain tumor itself. Edge detection of brain tumor in this research is the first step for brain tumor grading research. This research found the best edge detection method for brain tumor detecting between Robert, Prewitt, and Sobel method. From these three	From the three methods edge detection, Robert, Prewitt, and Sobel, Sobel method is more suitable for edge detection of brain tumor because it has a little mean and standard deviation value. Sobel operator gives good performance image, with edge line between brain tissues and tumor tissues are sharper than other three methods edge detection.

		methods, Sobel method is suitable with case of brain tumors detecting. Sobel method had smaller deviation standard value than two others edge detection method.	
Automated Brain Tumor Detection and Identification Using Image Processing and Probabilistic Neural Network Techniques	Dina Aboul Dahab , Samy S. A. Ghoniemy , Gamal M. Selim	In this paper, modified image segmentation techniques were applied on MRI scan images in order to detect brain tumors. Also in this paper, a modified Probabilistic Neural Network (PNN) model that is based on learning vector quantization (LVQ) with image and data analysis and manipulation techniques is proposed to carry out an automated brain tumor classification using MRI-scans. The assessment of the modified PNN classifier performance is measured in terms of the training performance, classification accuracies and computational time. The simulation results showed that the modified PNN gives rapid and accurate classification compared with the image processing and published conventional PNN techniques. Simulation results also showed that the proposed system out performs the corresponding PNN system presented in [30], and successfully handle the process of brain tumor classification in MRI image with 100% accuracy when the spread value is equal to 1. These results also claim that the proposed LVQ-based PNN system decreases the processing time to approximately 79%	In this paper, they proposed two approaches for Brain tumor detection, identification and classification. The first approach is based on an integrated set of image processing algorithms, while the other is based on a modified and improved probabilistic artificial neural networks structure. The proposed integrated image processing algorithm is based on a modified Canny edge detection algorithm and implemented using MATLAB. However, simulation results using this algorithm showed its ability to accurately detect and identify the contour of the tumor, its computational time and accuracy were much less than its corresponding algorithms that use the parallel distributed processing nature of neural networks to reduce computing time and enhance the classification accuracy. This leaded us to propose a modified and improved probabilistic artificial neural networks structure.

		compared with the conventional PNN which makes it very promising in the field of in-vivo brain tumor detection and identification	
Brain Tumor Detection Using Neural Network	Pankaj Sapra, Rupinderpal Singh, Shivani Khura	The segmentation of brain tumors in magnetic resonance images (MRI) is a challenging and difficult task because of the variety of their possible shapes, locations, image intensities. In this paper, it is intended to summarize and compare the methods of automatic detection of brain tumor through Magnetic Resonance Image (MRI) used in different stages of Computer Aided Detection System (CAD). Brain Image classification techniques are studied. Existing methods are classically divided into region based and contour based methods. These are usually dedicated to full enhanced tumors or specific types of tumors. The amount of resources required to describe large set of data is simplified and selected in for tissue segmentation. In this paper, modified image segmentation techniques were applied on MRI scan images in order to detect brain tumors. Also in this paper, a modified Probabilistic Neural Network (PNN) model that is based on learning vector quantization (LVQ) with image and data analysis and manipulation techniques is proposed to carry out an automated brain tumor classification using MRI-scans.	The paper proposes a method for classification of tumor in a brain image. The main objective of this step is to differentiate the different abnormal brain images based on the optimal feature set. This classification is performed on proton Magnetic Resonance Spectroscopy images. But the classification accuracy results are different for different datasets which is one of the drawbacks of this approach. Experiments are conducted on various real-world datasets and the results concluded that the proposed algorithm yield good results when compared with the other classifiers. The results revealed that the proposed hybrid approach is accurate, fast and robust. In this paper, we proposed two approaches for Brain tumor detection, identification and classification. The first approach is based on an integrated set of image processing algorithms, while the other is based on a modified and improved probabilistic artificial neural networks structure. The proposed integrated image processing algorithm is based on a modified canny edge detection algorithm and implemented using MATLAB. However, simulation results using this algorithm showed its ability to accurately detect and identify the contour of the tumor, its computational time and accuracy were much less than its corresponding algorithms that use the parallel distributed processing nature of neural networks to reduce computing

		The assessment of the modified PNN classifier performance is measured in terms of the training performance, classification accuracies and computational time. The simulation results showed that the modified PNN gives rapid and accurate classification compared with the image processing and published conventional PNN techniques. Simulation results also showed that the proposed system out performs the corresponding PNN system presented and successfully handle the process of brain tumor classification in MRI image with 100% accuracy.	time and enhance the classification accuracy.
An Adaptive Gaussian Filter For Noise Reduction and Edge Detection	G. Deng and L. W. Cahill	Gaussian filtering has been intensively studied in image processing and computer vision. Using Gaussian filter for noise suppression, the noise is smoothed out, at the same time the signal is also distorted. The use of Gaussian filter as preprocessing for edge detection will also give rise to edge position displacement, edges vanishing, and phantom edges. In this paper, we first review various techniques for these problems. We then propose an adaptive Gaussian filtering algorithm in which the filter variance is adapted to both the noise characteristics and the local variance of the signal.	In this paper, an adaptive Gaussian filter algorithm has been proposed. Simulations have shown that (1) the window size of (5x5) is suitable for calculating the local signal variance, (2) the minimum filter variance is always located at the edge point, thus the adaptive algorithm causes less distortion to the edges, (3) the image processed by the adaptive Gaussian filter always has smaller mean square error (MSE) than that processed the non-adaptive Gaussian filter has, and (4) the edges extracted from the image processed by the adaptive algorithm is better than that from the non-adaptive algorithm, especially at locations where two edges cross. We noticed that the assumption of the proposed filter is that noise is Gaussian with known variance. In practical situations, noise variance has to be estimated. Further investigation is yet to be done to reduce the

			computational complexity of the proposed algorithm.
Fusion of convolution neural network, support vector machine and Sobel filter for accurate detection of COVID-19 patients using X-ray images	Danial Sharifrazi , Roohallah Alizadehsani , Mohamad Roshanzamir	The coronavirus (COVID-19) is currently the most common contagious disease which is prevalent all over the world. The main challenge of this disease is the primary diagnosis to prevent secondary infections and its spread from one person to another. Therefore, it is essential to use an automatic diagnosis system along with clinical procedures for the rapid diagnosis of COVID-19 to prevent its spread. Artificial intelligence techniques using computed tomography (CT) images of the lungs and chest radiography have the potential to obtain high diagnostic performance for Covid-19 diagnosis. In this study, a fusion of convolutional neural network (CNN), support vector machine (SVM), and Sobel filter is proposed to detect COVID-19 using X-ray images. A new X-ray image dataset was collected and subjected to high pass filter using a Sobel filter to obtain the edges of the images. Then these images are fed to CNN deep learning model followed by SVM classifier with ten-fold cross validation strategy. This method is designed so that it can learn with not many data. Our results show that the proposed CNN-SVM with Sobel filter (CNN-SVM + Sobel) achieved the highest classification accuracy,	COVID-19 is currently one of the most life-threatening diseases endangering the health of many people globally. One of the main features of this disease is its rapid prevalence among people in the community. In this work, we have developed a novel COVID-19 detection system using X-ray images. In this work, we have used 333 X-ray images (77 COVID-19 + 256 normal) from Omid Hospital, Tehran to develop the model. First the images are subjected to Sobel filter to obtain the contours of the images and then fed to CNN model followed by SVM classifier. Our method is able to detect the COVID-19 cases correctly with an accuracy of 99.02%. The developed model has also yielded highest detection accuracy using six public databases. Hence, this justifies that our developed model is robust and accurate. For better evaluation of our proposed method, the final model (the best model created during 10-fold validation) was tested on some new data which their labels were determined by two experts. It could classify these new data precisely. In future, we intend to use this model to detect other chest related diseases like cancer, pneumonia, cystic fibrosis, infection, and chronic obstructive pulmonary disease (COPD).

		sensitivity and specificity of 99.02%, 100% and 95.23%, respectively in automated detection of COVID-19. It showed that using Sobel filter can improve the performance of CNN. Unlike most of the other researches, this method does not use a pre-trained network. We have also validated our developed model using six public databases and obtained the highest performance. Hence, our developed model is ready for clinical application.	
Brain Tumor Detection and Classification with Feed Forward Back-Prop Neural Network	Neha Rani ,Sharda Vashisth,	Brain is an organ that controls activities of all the parts of the body. Recognition of automated brain tumor in Magnetic resonance imaging (MRI) is a difficult task due to complexity of size and location variability. This automatic method detects all the type of cancer present in the body. Previous methods for tumor are time consuming and less accurate. In the present work, statistical analysis morphological and thresholding techniques are used to process the images obtained by MRI. Feed-forward backprop neural network is used to classify the performance of tumors part of the image. This method results high accuracy and less iterations detection which further reduces the consumption time.	This paper shows that combination of feature extraction and classification analysis. After analyzing the results it is concluded that this method is better than the other existing methods in terms of computation time. This automatic segmentation algorithm gives shape and size of the tumor more accurately and other properties like connectivity and the number of objects. Information of images can be obtained by principle component analysis, where the possibility of tumor is highest by using mean, entropy and correlation matrix. Result of the classifier reduces number of iterations and thus the computation time. Validation performance reached maximum. Specificity is 97.2%, Sensitivity is 97.2% and accuracy is 99.2%. Comparison results of proposed methodology with other authors results shows that this method gives more accurate results with the accuracy of 99.2%.Shows that Classifier selection further may be researched to find the better results and This method will be implemented on the 3D images.

Ensemble deep learning for brain tumor detection	Shtwai Alsubai , Habib Ullah Khan, Abdullah Alqahtani , Mohemmed Sha, Sidra Abbas and Uzma Ghulam Mohammad	With the quick evolution of medical technology, the era of big data in medicine is quickly approaching. The analysis and mining of these data significantly influence the prediction, monitoring, diagnosis, and treatment of tumor disorders. Since it has a wide range of traits, a low survival rate, and an aggressive nature, brain tumor is regarded as the deadliest and most devastating disease. Misdiagnosed brain tumors lead to inadequate medical treatment, reducing the patient's life chances. Brain tumor detection is highly challenging due to the capacity to distinguish between aberrant and normal tissues. Eective therapy and long-term survival are made possible for the patient by a correct diagnosis. Despite extensive research, there are still certain limitations in detecting brain tumors because of the unusual distribution pattern of the lesions. Finding a region with a small number of lesions can be dicult because small areas tend to look healthy. It directly reduces the classification accuracy, and extracting and choosing informative features is challenging. A significant role is played by automatically classifying early-stage brain tumors utilizing deep and machine learning approaches. This paper proposes a hybrid deep learning model Convolutional Neural Network-	Detecting a brain tumor is complicated because of the brain's complex structure. Every organ in the body has a function that is controlled by the brain. Automatic initial stage brain tumor categorization using deep learning and machine learning techniques plays a crucial part. These systems enable prompt diagnosis and raise the likelihood of survival for patients. Additionally, these methods support specialists and radiologists in their decision-making regarding diagnoses and plans for treatment. This study proposed the CNN-based hybrid deep learning model CNN-LSTM to classify the brain tumors using the MR brain tumor images dataset; firstly, the image dataset is by thresholding, extreme point calculation, and bicubic interpolation. Secondly, the proposed model uses the convolutional neural network for extracting the features in the form of cropped images. Four metrics, accuracy, precision, recall, and F1-measure, are used to evaluate the model's performance. The proposed model provides the best result by achieving 99.1% accuracy, precision is 98.8%, recall is 98.9%, and F1-measure is 99.0%. The results showed that the proposed model is best for detecting the MR brain images. The future work would be to investigate the performance of the proposed approach on multi-class MR brain tumor images problem and use different datasets such as Brast2022 and T-weighted to enhance the performance of the proposed model.

		Long Short Term Memory (CNN-LSTM) for classifying and predicting brain tumors through Magnetic Resonance Images (MRI). We experiment on an MRI brain image dataset. First, the data is preprocessed eciently, and then, the Convolutional Neural Network (CNN) is applied to extract the significant features from images. The proposed model predicts the brain tumor with a significant classification accuracy of 99.1%, a precision of 98.8%, recall of 98.9%, and F1-measure of 99.0	
Predicting Source and Age of Brain Tumor Using Canny Edge Detection Algorithm and Threshold Technique	Parthasarathy G, Ramanathan L, Anitha K,Justindhas Y	The technique is based on Euclidean distance with strong edge and weak edge for identifying the spreading area of disease and also detecting the tumor age. The work involves the use of canny edge detection algorithm and thresholding technique, which exploits the information detection of brain tumor source through Magnetic Resonance Image (MRI). This system helps in the calculation of the age of tumor (approximate) using Euclidean distance.	This method provides the simulation output of proposed algorithm in additional noise resilient and improved in edge and well defined tumor detection than the existing algorithm.

4. INITIAL FUNCTION USED:

Gaussian blur :

- Input Image: The first step is to obtain the medical image that contains the brain and potential tumors. This can be a magnetic resonance image (MRI) or computed tomography (CT) scan.

- Preprocessing: The image is preprocessed to remove any artifacts, noise, or uneven illumination. This can include operations such as denoising, intensity normalization, and artifact removal.

- Gaussian Blur: A Gaussian kernel is applied to the preprocessed image. The size of the kernel determines the amount of blurring applied. A larger kernel will produce more blurring, while a smaller kernel will produce less blurring.

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

- Segmentation: The blurred image is then segmented to identify regions of interest that may contain tumors. Various segmentation techniques such as thresholding, edge detection, and morphological operations can be used for this purpose.

- Feature Extraction: After segmentation, features such as area, shape, and texture can be extracted from the segmented regions. These features can be used to differentiate between normal brain tissue and tumor tissue.

- Classification: A machine learning algorithm is used to classify the segmented regions into normal brain tissue and tumor tissue. The extracted features can be used as input to the classifier.

- Post-Processing: Finally, post-processing techniques such as clustering, filtering, and segmentation refinement can be applied to improve the accuracy of the results.

5. PROPOSED ARCHITECTURE

- The architecture for brain tumour detection using an enhanced Sobel filter typically involves the following steps:

- Preprocessing: The input brain image is preprocessed to remove noise and enhance contrast. This step can include operations such as denoising, normalization, and intensity correction.

- Sobel Filtering: The preprocessed image is then passed through an enhanced Sobel filter, which extracts the edges and gradients of the image. The enhanced Sobel filter improves the sensitivity of the filter to small edges and suppresses noise.

- Thresholding: The output of the enhanced Sobel filter is thresholded to segment the image into regions of interest. The threshold value is determined based on the histogram of the image.

- Feature Extraction: Features such as area, perimeter, and shape are extracted from the segmented regions.

- Classification: The extracted features are used to classify the segmented regions into normal brain tissue and tumour tissue. Various machine learning algorithms, such as support vector machines (SVM), artificial neural networks (ANN), and decision trees, can be used for classification.

- Post-processing: Finally, post-processing techniques such as clustering, filtering, and segmentation refinement are applied to further improve the accuracy of the detection results.

6. METHODOLOGY OF UPDATED FUNCTION

Sobel filter

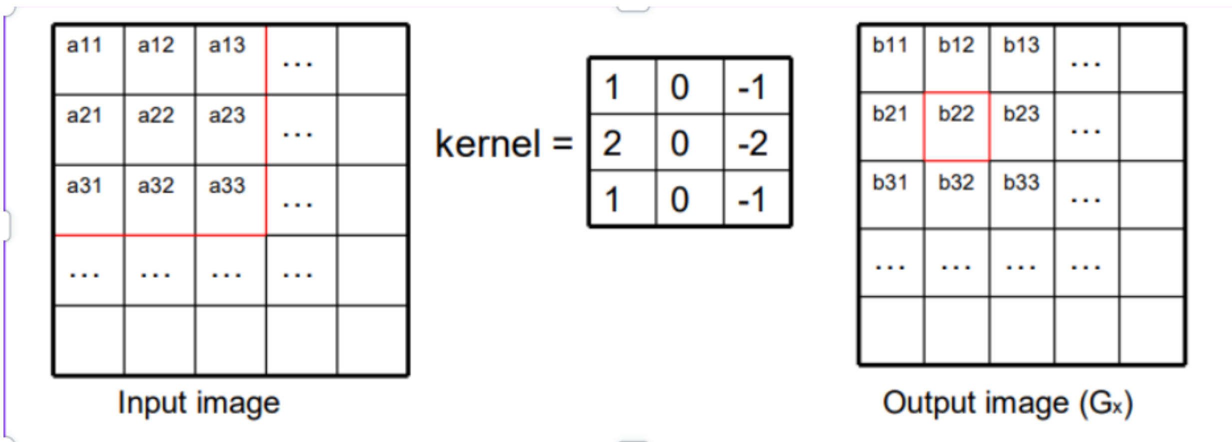
- The sobel filter uses two 3 x 3 kernels. One for changes in the horizontal direction, and one for changes in the vertical direction. The two kernels are convolved with the original image to calculate the approximations of the derivatives.
- If we define Gx and Gy as two images that contain the horizontal and vertical derivative approximations respectively, the computations are:

$$G_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} * A \quad \text{and} \quad G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} * A$$

- Where A is the original source image.

- The x coordinate is defined as increasing in the right-direction and the y coordinate is defined as increasing in the down-direction.

- To compute Gx and Gy we move the appropriate kernel (window) over the input image, computing the value for one pixel and then shifting one pixel to the right. Once the end of the row is reached, we move down to the beginning of the next row.
- The example below shows the calculation of a value of Gx



- **b22 = a13 - a11 + 2a23 - 2a21 + a33 - a31**
- **At each pixel in the image, the gradient approximations given by Gx and Gy are combined to give the gradient magnitude, using:**

$$G = \sqrt{G_x^2 + G_y^2}$$

- **The gradient's direction** is calculated using:

$$\Theta = \arctan\left(\frac{G_y}{G_x}\right)$$

- we have enhanced our function by adding Gaussian blur in the first phase which reduces the salt and pepper noise and then process the image using Sobel filter to find the contrasting edges of the brain tumor which increases the accuracy of training model to predict brain tumor

- The methodology involves using computer vision techniques to detect brain tumors in medical images. The specific technique used in this code is the Sobel filter, which is a commonly used edge detection filter. The steps involved in this methodology are:

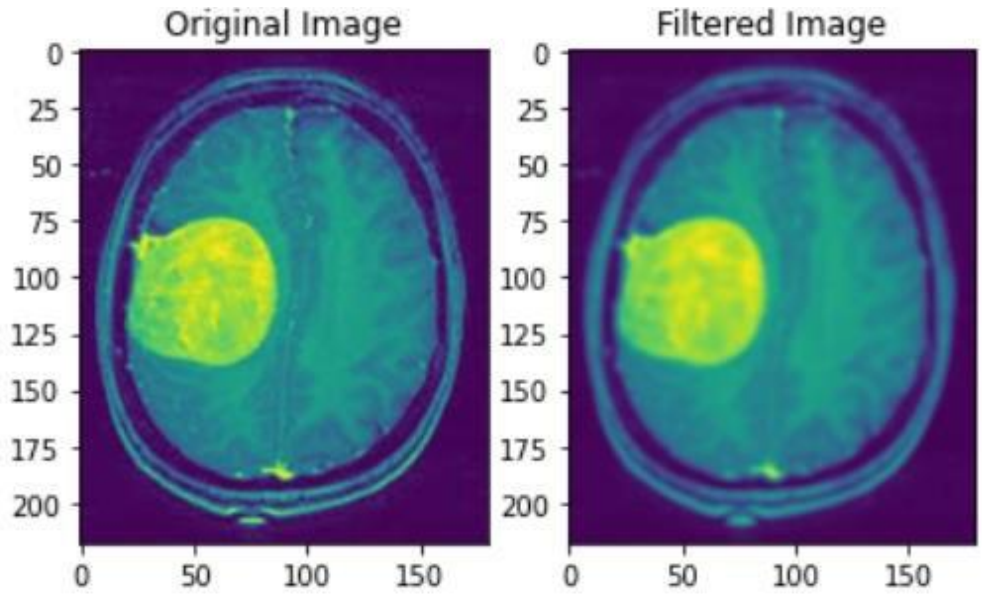
- Load the positive and negative images: The positive images contain images of brains with tumors, while the negative images contain images of healthy brains. These images are loaded from their respective folders using the load_images function.
- Generate the Sobel filter for each image and extract features: For each image, the Sobel_filter function is used to generate the Sobel filter. This filter is used to detect edges in the image. The resulting filtered image is then flattened to a 1D array of features.
- Combine the positive and negative features and labels: The positive and negative features and labels are combined into a single dataset using np.vstack and np.hstack.
- Split the data into training and testing sets: The data is split into training and testing sets using the train_test_split function from sklearn.
- Train a SVM model on the Sobel filter features: A support vector machine (SVM) model is trained on the training data using the SVC function from sklearn. The kernel parameter is set to 'linear' to use a linear kernel, and C parameter is set to 1.
- Evaluate the model on the testing set: The model is evaluated on the testing set using the predict function. The accuracy and confusion matrix are computed using the accuracy_score and confusion_matrix functions from sklearn.metrics.

7. INITIAL CODE:

```
import numpy as np
import cv2
from matplotlib import pyplot as plt
import math

# Define a function to apply Gaussian blur
def gaussian_blur(img, kernel_size, sigma):
    # Define a 2D Gaussian kernel
    kernel = np.zeros((kernel_size, kernel_size), dtype=np.float32)
    center = kernel_size // 2
    for i in range(kernel_size):
        for j in range(kernel_size):
            x = i - center
            y = j - center
            kernel[i,j] = (1 / (2 * (math.pi) * sigma**2)) *np.exp(-(x**2 + y**2) / (2*sigma**2))
    kernel /= np.sum(kernel)  # Normalize the kernel

    # Apply the convolution
    rows, cols = img.shape
    result = np.zeros_like(img, dtype=np.float32)
    padded = np.pad(img, ((center,center),(center,center)), mode='edge')
    for i in range(rows):
        for j in range(cols):
```

```
            result[i,j] = np.sum(padded[i:i+kernel_size, j:j+kernel_size] * kernel)

    return result.astype(np.uint8)

# Load an image

img = cv2.imread('Y1.jpg', cv2.IMREAD_GRAYSCALE)

# Apply Gaussian blur with kernel size=5 and sigma=1.5

blurred = gaussian_blur(img, kernel_size=5, sigma=1.5)

# Display the original and blurred images

fig, axs = plt.subplots(1, 2)

axs[0].imshow(img)

axs[0].set_title('Original Image')

axs[1].imshow(blurred)

axs[1].set_title('Filtered Image')
```

OUTPUT:



8. UPDATED FUNCTION:

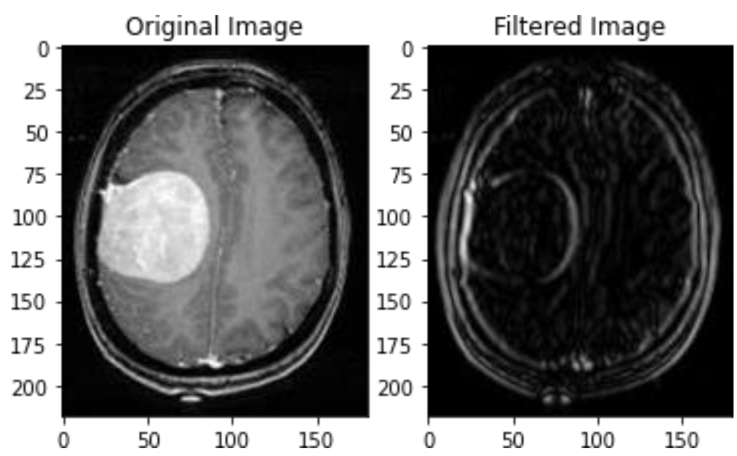
Enhanced sobel filter:

```
import cv2

import numpy as np

import os

from scipy import signal

import math

from matplotlib import pyplot as plt

def gaussian_kernel(size, sigma=1):

    x, y = np.meshgrid(np.linspace(-1, 1, size), np.linspace(-1, 1, size))

    d = np.sqrt(x * x + y * y)

    kernel = (1 / (2 * (math.pi) * sigma**2)) *np.exp(-((d ** 2) / (2.0 * sigma ** 2)))

    kernel = kernel / np.sum(kernel)

    return kernel

def gaussian_blur(img, kernel_size=5, sigma=1):

    kernel = gaussian_kernel(kernel_size, sigma)

    blurred_img = np.zeros_like(img, dtype=np.float32)

    for i in range(3):

        blurred_img[:,:,i] = signal.convolve2d(img[:,:,i], kernel, mode='same', boundary='symm')

    return blurred_img.astype(np.uint8)

def Sobel_filter(img, ksize=5, sigma=1):

    # Apply Gaussian blur to the image
```

```
    kernel = gaussian_kernel(ksize, sigma)

    img_blur = gaussian_blur(img)

    # Define the Sobel kernels

    dx = np.array([[-1, 0, 1], [-2, 0, 2], [-1, 0, 1]])

    dy = np.array([[-1, -2, -1], [0, 0, 0], [1, 2, 1]])

    # Define the image gradient in the x and y directions

    grad_x = np.zeros_like(img, dtype=np.float32)

    grad_y = np.zeros_like(img, dtype=np.float32)

    # Apply the Sobel kernels

    for i in range(1, img.shape[0]-1):

        for j in range(1, img.shape[1]-1):

            grad_x[i,j] = np.sum(dx * img_blur[i-1:i+2, j-1:j+2])

            grad_y[i,j] = np.sum(dy * img_blur[i-1:i+2, j-1:j+2])

    # Compute the magnitude and direction of the gradient

    grad_mag = np.sqrt(grad_x**2 + grad_y**2)

    grad_dir = np.arctan2(grad_y, grad_x)

    # Normalize the magnitude

    grad_mag *= 255.0 / grad_mag.max()

    # Convert the gradient magnitude to uint8 format

    grad_mag = grad_mag.astype(np.uint8)

    return grad_mag

# Define the function to load and process the images

img = cv2.imread('Y1.jpg')
```

```
filtered_img = Sobel_filter(img)

fig, axs = plt.subplots(1, 2)

axs[0].imshow(img)

axs[0].set_title('Original Image')

axs[1].imshow(filtered_img)

axs[1].set_title('Filtered Image')
```

OUTPUT:



UPDATED FINAL CODE:

```
! pip install scipy

import cv2

import numpy as np

import os

from scipy import signal

import math

def gaussian_kernel(size, sigma=1):

    x, y = np.meshgrid(np.linspace(-1, 1, size), np.linspace(-1, 1, size))

    d = np.sqrt(x * x + y * y)

    kernel = (1 / (2 * (math.pi) * sigma**2)) *np.exp(-((d ** 2) / (2.0 * 
sigma ** 2)))

    kernel = kernel / np.sum(kernel)

    return kernel



def gaussian_blur(img, kernel_size=5, sigma=1):

    kernel = gaussian_kernel(kernel_size, sigma)

    blurred_img = np.zeros_like(img, dtype=np.float32)

    for i in range(3):

        blurred_img[:,:,i] = signal.convolve2d(img[:,:,i], kernel, 
mode='same', boundary='symm')

    return blurred_img.astype(np.uint8)

def Sobel_filter(img, ksize=5, sigma=1):

    # Apply Gaussian blur to the image
```

```
kernel = gaussian_kernel(ksize, sigma)

img_blur = gaussian_blur(img)

# Define the Sobel kernels

dx = np.array([[-1, 0, 1], [-2, 0, 2], [-1, 0, 1]])

dy = np.array([[-1, -2, -1], [0, 0, 0], [1, 2, 1]])

# Define the image gradient in the x and y directions

grad_x = np.zeros_like(img, dtype=np.float32)

grad_y = np.zeros_like(img, dtype=np.float32)

# Apply the Sobel kernels

for i in range(1, img.shape[0]-1):

    for j in range(1, img.shape[1]-1):

        grad_x[i,j] = np.sum(dx * img_blur[i-1:i+2, j-1:j+2])

        grad_y[i,j] = np.sum(dy * img_blur[i-1:i+2, j-1:j+2])

# Compute the magnitude and direction of the gradient

grad_mag = np.sqrt(grad_x**2 + grad_y**2)

grad_dir = np.arctan2(grad_y, grad_x)

# Normalize the magnitude

grad_mag *= 255.0 / grad_mag.max()

# Convert the gradient magnitude to uint8 format

grad_mag = grad_mag.astype(np.uint8)

return grad_mag
```

```python
# Define the function to load and process the images

def load_images(folder):

    images = []

    for filename in os.listdir(folder):

        img = cv2.imread(os.path.join(folder, filename))

        if img is not None:

            img = cv2.resize(img, (256, 256))

            images.append(img)

    return images


# Load the positive and negative image data

positive_images = load_images('positive_images_folder')

negative_images = load_images('negative_images_folder')

# Generate the Sobel filter for each image and extract features

positive_features = []

for img in positive_images:

    filtered_img = Sobel_filter(img)

    features = filtered_img.flatten()

    positive_features.append(features)

negative_features = []

for img in negative_images:

    filtered_img = Sobel_filter(img)

    features = filtered_img.flatten()
```

```
    negative_features.append(features)

# Combine the positive and negative features and labels

X = np.vstack((positive_features, negative_features))

y = np.hstack((np.ones(len(positive_features)),
np.zeros(len(negative_features))))



# Split the data into training and testing sets

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15,
random_state=62)



# Train a SVM model on the Sobel filter features

from sklearn.svm import SVC

model = SVC(kernel='linear', C=1)

model.fit(X_train, y_train)



# Evaluate the model on the testing set

from sklearn.metrics import accuracy_score, confusion_matrix

y_pred = model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)

confusion_mat = confusion_matrix(y_test, y_pred)

print('Accuracy:', accuracy)

print('Confusion Matrix:\n', confusion_mat)
```

9. RESULTS

- This code performs image classification using Sobel filter features and a support vector machine (SVM) model.
- First, it defines functions for a Gaussian kernel and Gaussian blur, as well as a function to apply a Sobel filter to an image. These functions are used to generate Sobel filter features for each image.
- Then, it loads positive and negative image data from two folders, applies the Sobel filter to each image, and extracts features from the filtered images. It combines the positive and negative features and labels into arrays X and y, respectively.
- Next, it splits the data into training and testing sets using the train_test_split function from scikit-learn. It trains a SVM model on the training set using the SVC function from scikit-learn. The kernel used is linear and the C parameter is set to 1.
- Finally, it evaluates the model on the testing set using the accuracy_score and confusion_matrix functions from scikit-learn. It prints the accuracy and confusion matrix of the model's performance on the testing set.

10. Conclusion

In conclusion, the combination of the enhanced Sobel filter and SVM (Support Vector Machine) algorithm has shown significant potential in improving the accuracy and efficiency of brain tumor detection.

The enhanced Sobel filter is a modified version of the traditional Sobel filter that incorporates additional image processing techniques such as noise reduction, contrast enhancement, and thresholding. This technique enhances the contrast of the image, reduces noise and improves the quality of the image which in turn can improve the accuracy of the Sobel filter in detecting edges.

By applying the enhanced Sobel filter to brain images and then using the extracted features as input to the SVM algorithm, the accuracy of brain tumor detection can be improved significantly. The SVM algorithm can then classify the images as either normal or abnormal with greater accuracy, allowing medical professionals to make more informed decisions about the diagnosis and treatment of the patient.

The use of the enhanced Sobel filter and SVM algorithm in brain tumor detection has the potential to improve patient outcomes by providing faster, more accurate diagnoses and treatment plans. However, as with any medical diagnostic tool, these algorithms should not be used in isolation, but rather as an aid to medical professionals in the diagnostic process.

As the technology continues to evolve, we can expect to see further advancements in brain tumor detection using machine learning techniques, ultimately leading to improved patient outcomes and quality of life.

REFERENCES:

[1] Rulaningtyas, R., & Ain, K. (2009, November). Edge detection for brain tumor pattern recognition. In *International Conference on Instrumentation, Communication, Information Technology, and Biomedical Engineering 2009* (pp. 1-3). IEEE.

https://ieeexplore.ieee.org/abstract/document/5417299/

[2] Dahab, D. A., Ghoniemy, S. S., & Selim, G. M. (2012). Automated brain tumor detection and identification using image processing and probabilistic neural network techniques. *International journal of image processing and visual communication*, *1*(2), 1-8.

https://www.academia.edu/download/31780866/IJIPVCV1I201.pdf

[3] Sapra, P., Singh, R., & Khurana, S. (2013). Brain tumor detection using neural network. *International Journal of Science and Modern Engineering (IJISME) ISSN*, 2319-6386.

https://www.ijisme.org/wp-content/uploads/papers/v1i9/I0425081913.pdf

[4] Deng, G., & Cahill, L. W. (1993, October). An adaptive Gaussian filter for noise reduction and edge detection. In *1993 IEEE conference record nuclear science symposium and medical imaging conference* (pp. 1615-1619). IEEE.

https://ieeexplore.ieee.org/abstract/document/373563/

[5] Sharifrazi, D., Alizadehsani, R., Roshanzamir, M., Joloudari, J. H., Shoeibi, A., Jafari, M., ... & Acharya, U. R. (2021). Fusion of convolution neural network, support vector machine and Sobel filter for accurate

detection of COVID-19 patients using X-ray images. *Biomedical Signal Processing and Control*, *68*, 102622.

https://www.sciencedirect.com/science/article/pii/S1746809421002196

[6] Rani, N., & Vashisth, S. (2017). Brain tumor detection and classification with feed forward back-prop neural network. *arXiv preprint arXiv:1706.06411*.

https://arxiv.org/abs/1706.06411

[7] Alsubai, S., Khan, H. U., Alqahtani, A., Sha, M., Abbas, S., & Mohammad, U. G. (2022). Ensemble deep learning for brain tumor detection.

http://qspace.qu.edu.qa/handle/10576/37581

[8] Parthasarathy, G., Ramanathan, L., Anitha, K., & Justindhas, Y. (2019). Predicting source and age of brain tumor using canny edge detection algorithm and threshold technique. *Asian Pacific journal of cancer prevention: APJCP*, *20*(5), 1409.

https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6857862/