

BRAIN TUMOR DETECTION USING PYTHON

¹Bhulakshmi, Shaik.Shahina, Shaik. Sumayya, D. Daniel Paul, G. Sunil babu

¹Assistant Professor, Department of CSE, Priyadarshini Institute of Technology & Science, Chintalapudi, Tenali, Andhra Pradesh , India

UG,Department of CSE, Priyadarshini Institute of Technology & Science, Chintalapudi, Tenali, Andhra Pradesh , India

ABSTRACT:

The tremendous success of machine learning algorithms at image recognition tasks in recent years intersects with a time of dramatically increased use of electronic medical records and diagnostic imaging. This review introduces the machine learning algorithms as applied to medical image analysis, focusing on convolutional neural networks, and emphasizing clinical aspects of the field. The advantage of machine learning in an era of medical big data is that significant hierarchal relationships within the data can be discovered algorithmically without laborious hand-crafting of features. We cover key research areas and applications of medical image classification, localization, detection, segmentation, and registration. We conclude by discussing research obstacles, emerging trends, and possible future directions.

INDEX TERMS : Convolutional neural networks, medical image analysis, machine learning, deep learning.

INTRODUCTION:

Machine learning algorithms have the potential to be invested deeply in all fields of medicine, from drug discovery to clinical decision making, significantly altering the way medicine is practiced. The success of machine learning algorithms at computer vision tasks in recent years comes at an opportune time when medical records are increasingly digitalized. The use of Electronic Health Records (EHR) quadrupled from 11.8% to 39.6% amongst of based physicians in the US from 2007 to 2012 [1]. Medical images are an integral part of a patient's EHR and are currently analysed by human radiologists, who are limited by speed, fatigue, and experience. It takes years and great financial cost to train a qualified radiologist, and some health-care systems outsource radiology reporting to lower- cost countries such as India via

tele-radiology. A delayed or erroneous diagnosis

causes harm to the patient. Therefore, it is ideal for medical image analysis to be carried out by an automated, accurate and efficient machine learning algorithm.

Brain tumors can be classified into two types: benign (noncancerous) and malignant (cancerous). The malignant tumors can quickly spread to other tissues in the brain and lead to destroyed and replaced by new cells. If damaged and old cells are not eliminated with generating the new cells, it can cause problems. The production of additional cells often results in the formation of a mass of tissue, which refers to the growth or tumor. Brain tumor detection is very complicated and difficult due to the size, shape, location and type of tumor in the brain. accurately measure the size and resolution of the tumor.

Related work:

Recently, Machine learning (ML) and Deep Learning (DL) methods are widely been used for detection and grading brain tumors using different imaging modalities, especially those acquired using MRI. In this section, the most recent and related research works on the paper topic are presented. Mohsen, Heba, et al. [8] propose a system that combines discrete wavelet transform (DWT) features and deep learning (DL) techniques.

Methodology

BRaTS MICCAI dataset:

The Multimodal Brain Tumor Segmentation (BRaTS) MICCAI has always been focusing on the evaluation of state-of-the-art methods for the segmentation of brain tumors in magnetic resonance imaging (MRI) scans. Ample multi-institutional routine clinically-acquired multimodal MRI scans of glioblastoma (GBM) and lower grade glioma (LGG), with pathologically confirmed diagnosis and available OS, was provided as the training, validation and testing data for BRaTS 2015 challenge.

In countries with poor health-care systems, a deep learning analytical framework can be a helpful alternative tool. We utilize transfer learning to train a deep CNN

with weights pretrained on ImageNet using a weighted loss function. The effectiveness of this approach is shown by quantitative results on the Brain Tumor dataset, which achieves an F1-score of 92% and classification accuracy of 92% on the test set. In the future, work will be done on a bigger dataset and with more pretrained models. Data Augmentation:

Data augmentation :

Data augmentation consists of Grey Scaling (RGB/BW to ranges of grey), Reflection (vertical / horizontal flip), Gaussian Blur (reduces image noise), Histogram Equalization (increases global contrast), Rotation (may not preserve image size), Translation (moving the image along vertical shifts, and horizontal and vertical flips). Finally, the convolution neural network is used for automatic brain tumor classification. The brain image dataset is taken from image net. Image net is a one of the pre-trained model. If you want to train from the starting layer, we have to train the entire layer (i.e) up to ending layer. The loss function calculation is very important to improve the accuracy. If the loss function is high, when the accuracy is low. Similarly, the accuracy is high, Then the loss function is low. The gradient value is calculated for loss function to compute gradient descent algorithm. Repeatedly evaluate the gradient value to compute the gradient of loss function.

Image Pre-Processing:

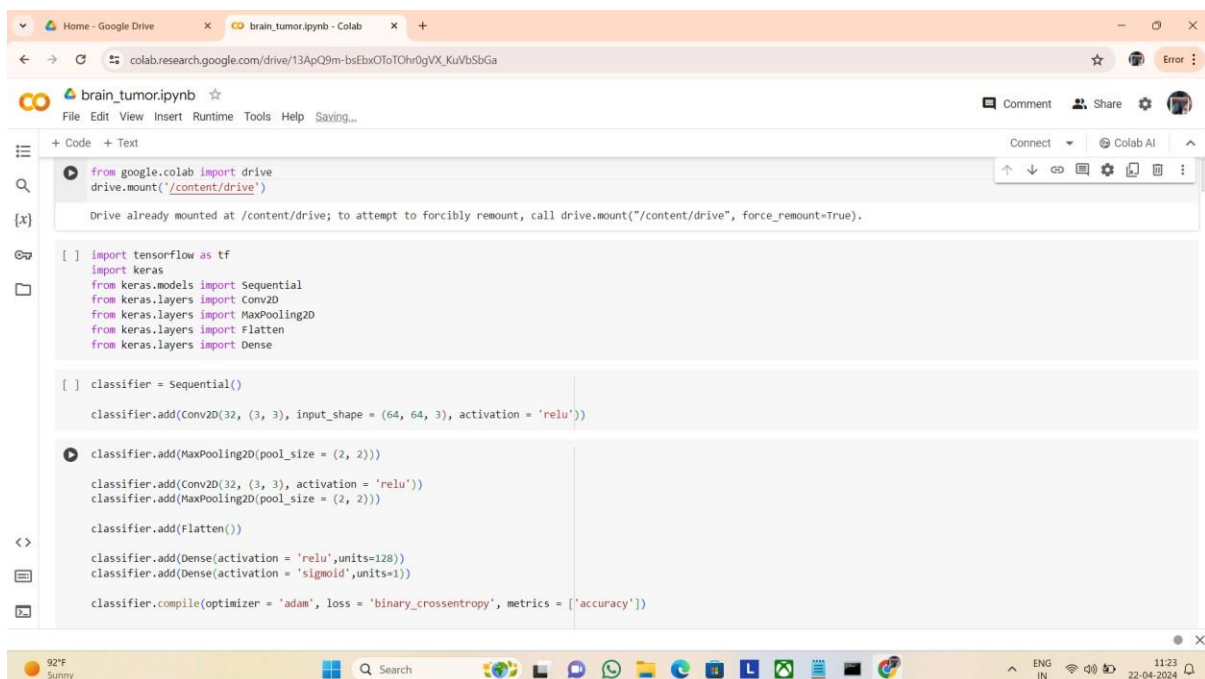
Our pre-processing includes rescaling, noise removal to enhance the image, applying

Binary Thresholding and morphological operations like erosion and dilation, contour forming (edge based methodology). In the first step of pre-processing, the memory space of the image is reduced by scaling the gray-level of the pixels in the range 0-255. We used Gaussian blur filter for noise removal as it is known to give better results than Median filter since the outline of brain is not segmented as tumor here

Segmentation:

Brain tumor segmentation involves the process of separating the tumor tissues (Region of Interest – ROI) from normal brain tissues and solid brain tumor with the help of MRI images or other imaging modalities. Its mechanism is based on identifying similar type of subjects the objects and group the objects having most similarity or finding the dissimilarity measure among the objects and separate the most dissimilar objects in the space.

Result:



```
from google.colab import drive
drive.mount('/content/drive')

Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

[ ] import tensorflow as tf
import keras
from keras.models import Sequential
from keras.layers import Conv2D
from keras.layers import MaxPooling2D
from keras.layers import Flatten
from keras.layers import Dense

[ ] classifier = Sequential()

classifier.add(Conv2D(32, (3, 3), input_shape = (64, 64, 3), activation = 'relu'))

classifier.add(MaxPooling2D(pool_size = (2, 2)))

classifier.add(Conv2D(32, (3, 3), activation = 'relu'))
classifier.add(MaxPooling2D(pool_size = (2, 2)))

classifier.add(Flatten())

classifier.add(Dense(activation = 'relu', units=128))
classifier.add(Dense(activation = 'sigmoid', units=1))

classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])
```

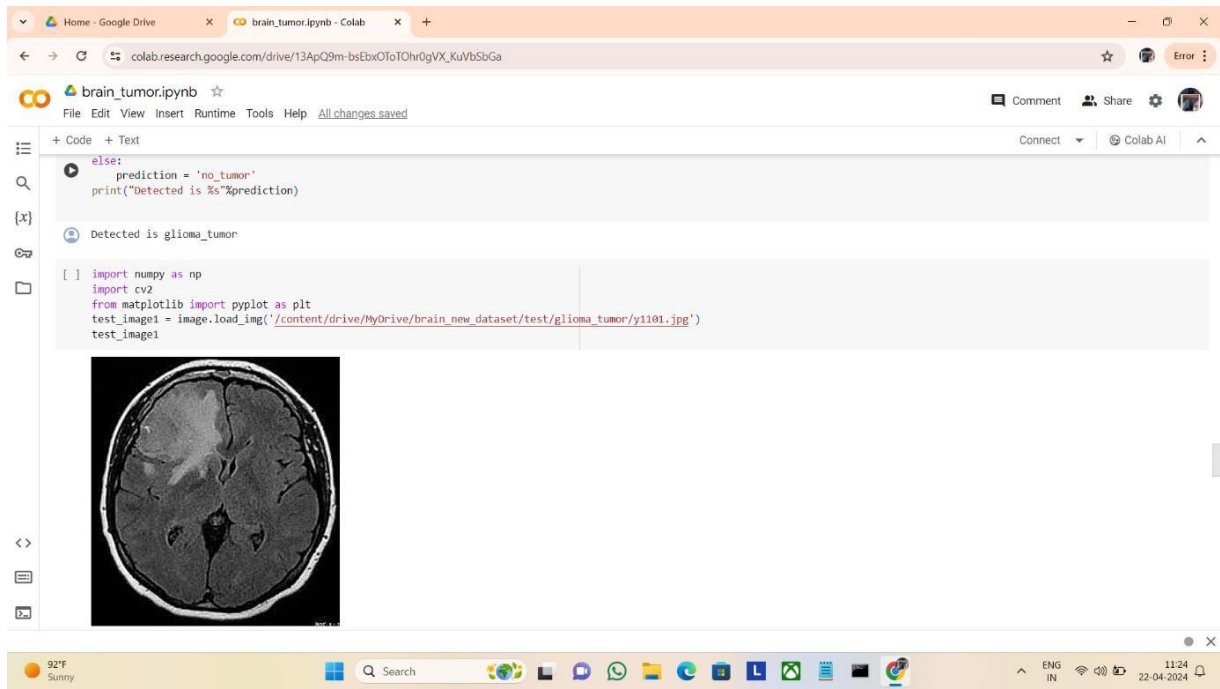
Found 2102 images belonging to 2 classes.
Found 898 images belonging to 2 classes.

```
history = classifier.fit_generator(training_set, steps_per_epoch=None, epochs=100, verbose=1, callbacks=None, validation_data=test_set, validation_steps=None, class_weight=None, max_c
```

Epoch	Time	Loss	Accuracy	Val Loss	Val Accuracy
66/66	22s 334ms/step	0.0121	0.9957	0.3045	0.9399
66/66	22s 335ms/step	0.0145	0.9948	0.2947	0.9577
66/66	23s 345ms/step	0.0126	0.9957	0.3167	0.9577
66/66	22s 335ms/step	0.0324	0.9914	0.4476	0.9254
66/66	23s 339ms/step	0.0391	0.9838	0.5055	0.9254
66/66	22s 339ms/step	0.0405	0.9867	0.2872	0.9532
66/66	23s 340ms/step	0.0166	0.9952	0.2470	0.9618
66/66	24s 361ms/step	0.0170	0.9943	0.3318	0.9521
66/66	23s 348ms/step	0.0062	0.9981	0.3085	0.9599
66/66	22s 336ms/step	0.0119	0.9962	0.4227	0.9376
66/66	24s 363ms/step	0.0126	0.9943	0.3462	0.9543
66/66	23s 353ms/step	0.0113	0.9962	0.3129	0.9555
66/66	24s 357ms/step	0.0199	0.9919	0.2775	0.9588
66/66	23s 353ms/step	0.0093	0.9971	0.2717	0.9599

```
plt.title('model accuracy')  
plt.ylabel('accuracy')  
plt.xlabel('epoch')  
plt.legend(['train', 'test'], loc='upper left')  
plt.show()
```

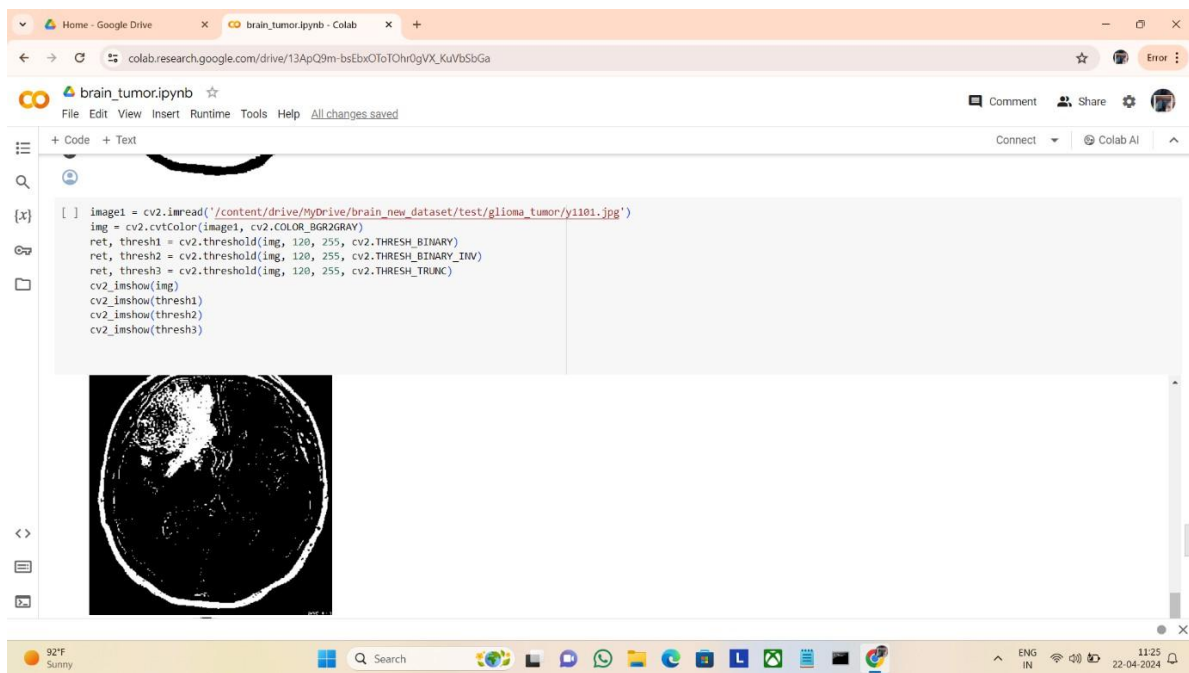
```
[ ] # summarize history for loss  
plt.plot(history.history['loss'])  
plt.plot(history.history['val_loss'])  
plt.title('model loss')  
plt.ylabel('loss')  
plt.xlabel('epoch')  
plt.legend(['train', 'test'], loc='upper left')  
plt.show()
```



The screenshot shows a Google Colab notebook titled 'brain_tumor.ipynb'. The code cell contains the following Python code:

```
else:  
    prediction = 'no_tumor'  
    print("Detected is %s"%prediction)  
  
[ ] import numpy as np  
import cv2  
from matplotlib import pyplot as plt  
test_image1 = image.load_img('/content/drive/MyDrive/brain_new_dataset/test/glioma_tumor/y1101.jpg')  
test_image1
```

The output of the code cell shows the text: "Detected is glioma_tumor". Below the code cell, a brain MRI image is displayed. The system tray at the bottom shows a temperature of 92°F, a search bar, and the date 22-04-2024.



The screenshot shows a Google Colab notebook titled 'brain_tumor.ipynb'. The code cell contains the following Python code:

```
[ ] image1 = cv2.imread('/content/drive/MyDrive/brain_new_dataset/test/glioma_tumor/y1101.jpg')  
img = cv2.cvtColor(image1, cv2.COLOR_BGR2GRAY)  
ret, thresh1 = cv2.threshold(img, 120, 255, cv2.THRESH_BINARY)  
ret, thresh2 = cv2.threshold(img, 120, 255, cv2.THRESH_BINARY_INV)  
ret, thresh3 = cv2.threshold(img, 120, 255, cv2.THRESH_TRUNC)  
cv2.imshow('img')  
cv2.imshow('thresh1')  
cv2.imshow('thresh2')  
cv2.imshow('thresh3')
```

Below the code cell, a binary brain MRI image is displayed, showing the tumor region in white against a black background. The system tray at the bottom shows a temperature of 92°F, a search bar, and the date 22-04-2024.

Conculsion:

Without pre-trained Keras model, the train accuracy is 97.5% and validation accuracy is 90.0%. The validation result had a best figure of 91.09% as accuracy. It is observed that without

using pre-trained Keras model, although the training accuracy is >90%, the overall accuracy is low unlike where pre-trained model is used. Also, when we trained our dataset without Transfer learning, the computation time was 40 min whereas when we used Transfer Learning, the computation time was 20min. Hence, training and computation time with pre-trained Keras model was 50% lesser than without. Chances over over-fitting the dataset is higher when training the model from scratch rather than using pre-trained Keras. Keras also provides an easy interface for data augmentation. Amongst the Keras models, it is seen that ResNet 50 has the best overall accuracy as well as F1 score. ResNet is a powerful backbone model that is used very frequently in many computer vision tasks..

Reference:

- [1] M. Karuna and A. Joshi, Automatic detection and severity analysis of brain tumors using gui in matlab, International Journal of Research in Engineering and Technology, 10, pp. 586-594, 2013.
- [2] KS. Aboody, A. Brown, et al, Neural stem cells display extensive tropism for pathology in adult brain Evidence from intracranial gliomas, Proceedings of the National Academy of Sciences, 97 (23), pp. 1284612851, 2000.
- [3] A. joshi, D. H. Shah, et al, Survey of brain tumor detection techniques through MRI images, International Journal of Research in Engineering and Technology, 10, pp. 586-594, 2013.

- [4] JP. Poonam, Review of image processing techniques for automatic detection of tumor in human brain, *International Journal of Computer Science and Mobile Computing*, 2(11), pp. 117-122, 2013.
- [5] H. Cecotti and A. Graeser, Convolutional neural network with embedded Fourier transform for EEG classification, *Pattern Recognition, 19th International Conference on. IEEE, ICPR*, pp. 1-14, 2008.
- [6] R. Bayot and T. Goncalves, A survey on object classification using convolutional neural networks, 2015.
- [7] M. Soltaninejad, et al, Automated brain tumour detection and segmentation using superpixel-based extremely randomized trees in FLAIR MRI, *International journal of computer assisted radiology and surgery*, 12(2), pp. 183-203, 2017.
- [8] S. Pereira, et al, Brain tumor segmentation using convolutional neural networks in MRI images, *IEEE transactions on medical imaging*, 35(5), pp. 1240- 1251, 2016.
- [9] Halimeh Siar, Mohammad Teshnehlab, Diagnosing and Classification Tumors and MS Simultaneous of Magnetic Resonance Images Using Convolution Neural Network, *7th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, 2019.
- [10] L. Szilagyi, et al, Automatic brain tumor segmentation in multispectral MRI volumes using a fuzzy c-means cascade algorithm, In *2015 12th international conference on fuzzy systems and knowledge discovery (FSKD)*, IEEE, pp. 285- 291, 2015.
- [11] Y. Xu, et al, Deep convolutional activation features for large scale brain tumor histopathology image classification and segmentation, In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 947-951, 2015.

- [12] Y. Pan, et al, Brain tumor grading based on neural networks and convolutional neural networks, In 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 699-702, 2015.
- [13] F. Samadi, G. Akbarizadeh, et al, Change Detection in SAR Images using Deep Belief Network: a New Training Approach based on Morphological Images, IET Image Processing, 2019.
- [14] M. Havaei, et al, Brain tumor segmentation with deep neural networks, Medical image analysis, 35, 18-31, 2017.
- [15] E. Alpaydin, Introduction to Machine Learning , London: The MIT Press. pp. 110, 2009.
- [16] W. Zhang, K. Itoh, et al, Parallel Distributed Processing Model with local Space-Invariant Interconnections and its Optical Architecture, Applied optics, 29(32), pp. 4790-4797, 1990.
- [17] LY. LeCun, K. Kavukcuoglu, et al, Convolutional networks and applications in vision, In ISCAS, pp. 253-256, 2010.
- [18] HH. Aghdam, EJ. Heravi et al, Guide to Convolutional Neural Networks, A Practical Application to Traffic-Sign Detection and Classification, Springer, 2017.
- [19] D. Ciresan, U. Meier, et al, Multi-column deep neural networks for image classification. Int.Conf. Computer Vision and Pattern Recognition, IEEE, pp. 1202.2745, 2012.
- .