ISSN: 2454-9940



INTERNATIONAL JOURNAL OF APPLIED SCIENCE ENGINEERING AND MANAGEMENT

E-Mail : editor.ijasem@gmail.com editor@ijasem.org





www.ijasem.org

Vol 18, Issue 1, 2024

AUTOMATED SEGMENTATION OF BRAIN TUMOR MRI IMAGES USING DEEP LEARNING

Ms. SK. NAGEENA, Assistant Professor,Department Of ECE, SICET, Hyderabad Prattipati Madhavi, Koyi Chandana Harshitha, V Sunil Kumar, Yadma Abhinay Reddy Department Of ECE, SICET, Hyderabad

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 inall 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 verycomplicated 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:



www.ijasem.org

Vol 18, Issue 1, 2024

Recently, Machine learning (ML) and Deep Learning (DL) methods are widely been used for etection and grading brain tumors using different imaging modalities, especially those acquired using MRI. In this section, the most recentand 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 alwaysbeen focusing on he 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 augumentation 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 ertical 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, Thenthe 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



www.ijasem.org

Vol 18, Issue 1, 2024

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:

| <pre> () () () () () () () () () () () () ()</pre> | - | | | | - | | | |
|--|-------------------------|--|-----------------|--------------------|--|--------------------------------|---|---------|
| <pre> Arring to the first the first the first. Arr of the first the first. Arr of the first the first the first. Arr of the first the first the first. Arr of the first the first the firs</pre> | > | O C colab.research.google.com/drive/13ApQ9m-bsEbxOToTOHr0gVX_KuVbSbGa | | ¢ 🖗 | Error | : | | |
| <pre>the text when the new we make the second text of the second text</pre> | 0 | 6 brain_tumor.ipynb ☆ | Comment | 🖹 Share 🏼 🛱 | - |) | | |
| <pre> for the section of the sectin of the section of the section of the sect</pre> | | + Code + Text | Connect | 👻 🚱 Colab A | I I I | ~ | | |
| if increased (//unit (/unit | h | from google.colab import drive | ↑ ↓ a | e 🛛 🗘 🗐 | | | | |
| Image: State and State an | | drive.mount('/ <u>content/drive</u> ') Delive sleady at (content/drive to streamt to feerblu persuat call deive mount/"/content/drive" force remount.True) | | | | | | |
| <pre></pre> | | urive aiready mounted at /content/urive; to attempt to forcibly remount, call urive.mount(/content/urive , force_remount=irue). | | | | | | |
| Image: Internet and image: State internet in | | [] import tensorflow as tf import keras | | | | | | |
| Image: top::::::::::::::::::::::::::::::::::: | | from keras.models import Sequential from keras.layers import Conv2D | | | | | | |
| I is instruction to bese [] i cissifier-signed cond(x); (),), ips(tables + (x, s)), ettotion + 'rele')); C is infier-signed cond(x); (),), ips(tables + (x, s)); cissifier-signed cond(x); cissifier-signe | | from keras.layers import MaxPooling20 from keras.layers import Flatten | | | | | | |
| [] | | from keras.layers import Dense | | | | | | |
| <pre>clasifie_adjoards[]; (;);); bar_tuber ((a, i,); extuation - 'nale')) (lasifie_adjoards[]; (;);); bar_tuber ((a, i,)); extuation - 'nale')) (lasifie_adjoards[]; (;);); bar_tuber ((a, i,)); (lasifie_adjoards[]; (;);); bar_tuber ((a, i,)); (lasifie_adjoards[]; (i,); bar_tuber ((a, i,)); (lasif</pre> | | <pre>[] classifier = Sequential()</pre> | | | | | | |
| c tastifier_adjopsic_(; r); j; j; dituitier (r); dituitier); dituitier (r); dituitier (r); dituitier (r) | | <pre>classifier.add(Conv2D(32, (3, 3), input_shape = (64, 64, 3), activation = 'relu'))</pre> | | | | | | |
| <pre>listifer_add(c):::::::::::::::::::::::::::::::::::</pre> | | <pre>classifier.add(MaxPooling2D(pool_size = (2, 2)))</pre> | | | | | | |
| <pre>distifier.edd()Inte()) distifier.edd()Inte()) distifier.edd()Inte()) distifier.complic(opticity = 'ndm', loss = 'blong, (researchy')) distifier.complic(opticity = 'ndm', loss = 'blong, (researchy)) distifier.complic(opticity = 'ndm', loss = 'blong, (researchy)) distifier.complic(opticity = 'ndm', loss = 'blong, (researchy)) distifier.complic(optic) * disting = 'ndm', loss = 'blong, (researchy) disting = 'ndm', loss = 'ndm', loss = 'blong, (researchy) disting = 'ndm', loss = 'ndm', loss</pre> | | <pre>classifier.add(Conv2D(32, (3, 3), activation = 'relu')) classifier.add(MaxPooline2D(nool size = (2, 2)))</pre> | | | | | | |
| <pre> ilistifier.ab(kensequetization = 'telu', attistig) ilistifier.ab(kensequetization = 'telu', attistig)</pre> | | classifier.add(Flatten()) | | | | | | |
| Chaiffer.dig(opticitivation - 'signif', units-1)) Chaiffer.complie(opticitize - 'adm', loss + 'binay_crossentropy', metrics - [accuracy']) Were Were One - complie None - complex × () Disarding complexity Image: compl | | <pre>classifier.add(Dense(activation = 'relu',units=128))</pre> | | | | | | |
| Classifier.complic(optimizer = 'safer', Loss = 'bilary_crosseetropy', metrics = ['accuracy']) ***** **** ***** ***** ***** ***** **** | | <pre>classifier.add(Dense(activation = 'sigmoid',units=1))</pre> | | | | | | |
| Winty Q. Sauch Wint Winty < | | <pre>classifier.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['accuracy'])</pre> | | | | | | |
| ² /2 ² / | | | | | 0 | × | | |
| File Edit View Insert Runtime Tools Help Allchanges.saxed Code + Text Connect Connec | Hom | swme - Google Drive X CO brain, tumor.ipymb - Colab X + CC :: colab/research.google.com/drive/13Ap/Q9m-bsEbx/OTO/Dr/UgVX_KuVbSbGa | IN | 22-04 | -2024 | - | - • | D E |
| Code + Text Connect • © Colab Al • fermid 2102 images belonging to 2 classes. • Connect • © Colab Al • history - classifier.fit_generator(training_set, steps_per_epoch=None, epoch=None, verbose=1, callbacks=None, validation_data=test_set, validation_steps=None, class_weight=None, r • bistory - classifier.fit_generator(training_set, steps_per_epoch=None, epoch=None, verbose=1, callbacks=None, val_accuracy: 0.9399 • Val_accuracy: 0.9399 • bistory - classifier.fit_generator(training_set, steps_per_epoch=None, epoch=None, verbose=1, callbacks=None, val_accuracy: 0.9399 • Val_accuracy: 0.9399 • bistory - classifier.fit_generator(training_set, steps_per_epoch=None, epoch=None, verbose=1, callbacks=None, val_accuracy: 0.9399 • Val_accuracy: 0.9399 • fpoch 73/100 - 225 335ms/step = loss: 0.0121 - accuracy: 0.9997 - val_loss: 0.3045 - val_accuracy: 0.9577 • fpoch 77/100 - 225 335ms/step = loss: 0.0126 - accuracy: 0.9997 - val_loss: 0.3167 - val_accuracy: 0.9577 • fpoch 77/100 - 225 335ms/step = loss: 0.0391 - accuracy: 0.9988 - val_loss: 0.4276 - val_accuracy: 0.9254 • fpoch 77/100 - 225 339ms/step = loss: 0.0166 - accuracy: 0.9997 - val_loss: 0.2270 - val_accuracy: 0.9512 • fpoch 77/100 - 225 336ms/step = loss: 0.0170 - accuracy: 0.9991 - val_loss: 0.3217 - val_accuracy: 0.9512 • fpoch 77/100 - 225 336ms/step = loss: 0.0126 - accuracy: 0.9962 - val_loss: 0.4227 - val_accuracy: 0.9512 • fpoch 8/100 | Hom | ammy amme - Google Drive X @ brain_tumor.lpynb - Colab X + C to colab.research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa I brain_tumor.ipynb 🔅 | IN | Comment | •2024 | - | 0 9 | D |
| Found 896 images belonging to 2 classes. history = classifier.fit_generator(training_set, steps_per_epoch=tone, epochs=100, verbose=1, callbacks=tone, validation_data=test_set, validation_steps=Hone, class_weight=Hone, r 66/66 [| Hom C S File | ome-Google Drive X CO brain_tumor.laynb - Colab X + C S colab.research.google.com/drive/13ApQ9m-bsEbxOToTOHr0gVX_KuVb5bGa brain_tumor.ipynb 🛠 e Edit View Insert Runtime Tools Help All.changes.saved | IN | Comment | 2024 | _ ☆ Share | 0 9 0 | 5) E |
| 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, r 6/66 [| Horr C Coc | ome-Google Drive X 00 brain_tumor.lpynb-Colab X + C C C Colab.research.google.com/drive/13ApQ9m-bsEbxOToTOHr0gVX_KuVbSbGa brain_tumor.ipynb ★ le Edit View Insert Runtime Tools Help All.changes.saved ode + Text Forum 2102, issues belonging to 2 classes | IN | Connect | 2024 | → Share © Col | c R S Ab Al | E |
| 6/6 6/6 [========] - 225 3348s/step - loss: 0.0121 - accuracy: 0.9997 - val_loss: 0.3045 - val_accuracy: 0.9399 6/66 [==========] - 225 334s/step - loss: 0.0145 - accuracy: 0.9997 - val_loss: 0.3045 - val_accuracy: 0.9577 100 6/66 [================================== | Horr C File | amme - Google Drive X 4 C Scolab.research.google.com/drive/13ApQ9m-bsEbxOToTOHr0gVX_KuVbSbGa b Drain_tumor.ipynb Image: Comparison of the com | IN | Connect | 2024 2. | → Share © Col | c R S ab Al | E |
| 6) 60 [=================================== | Hon C File Coo | ame-Google Drive X OP brain_tumor/pymb-Colab X + C C colab/research.google.com/drive/13Ap.Q9m-bsEbxOToTOHr0gVX_KuVbSbGa brain_tumor.ipymb C E Edit View Insert Runnime Tools Help All_changes.saved ode + Text * Found 2102 images belonging to 2 classes. Found 898 images belonging to 2 classes. history = classifier.fit_generator(training_set, steps_per_epoch=None, epoch=100, verbose=1, callbacks=None, validation_data=test | set, validation | Comment Connect | 2024 • | - Share ⊚ Col | c P ab Al | 9 E |
| bobbs i==================================== | Hon C File Coc | summy X OP brain_tumor.ipynb - Colab X + C Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: brain_tumor.ipynb Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Image: colab/research.google.coom/drive/13ApQ9m-bsEbxO | set, validation | Comment Connect | * :lass_ | → Share © Col weight= | c P ab Al | 9 E |
| 00/06 [==================================== | Hon C File Coc | <pre>smm; smm; smm; C sclabsresearch.google.com/drive/13ApQ9m-bsEbxOToTOHr0gVX_KuVbSbGa brain_tumor.ipynb ☆ le Edit View Insert Runtime Tools Help All.changes.saved ode + Text Found 2102_images belonging to 2 classes. Found 808_images belonging to 2 classes.</pre> | set, validation | Connect | 2. Class_ | Share ⊚ Col weight= | c P ab Al Ione, | 9 E |
| b0/00 [-/100 b0/00 [-/100 b0/06 [-/100 b0/100 [| Hon C File Coc | <pre>summy summy summy</pre> | set, validation | Connect | Class_ | → Share © Col weight= | c P ab Al Vone, |) (B |
| by: 1/100 66/66 1-225 339ms/step - 1025 0.0025 - accuracy: 0.9806 - val_055: 0.2027 - val_accuracy: 0.9532 66/66 1-205 340ms/step - 1025: 0.0166 - accuracy: 0.9986 - val_055: 0.2027 - val_accuracy: 0.9532 66/66 1-205 340ms/step - 1055: 0.0166 - accuracy: 0.9952 - val_055: 0.2470 - val_accuracy: 0.9510 66/66 1-205 340ms/step - 1055: 0.0170 - accuracy: 0.9992 - val_055: 0.2470 - val_accuracy: 0.9511 66/66 1-205 340ms/step - 1055: 0.0170 - accuracy: 0.9992 - val_055: 0.2470 - val_accuracy: 0.9521 66/66 1-205 340ms/step - 1055: 0.0170 - accuracy: 0.9991 - val_055: 0.3085 - val_accuracy: 0.9599 66/66 1-215 340ms/step - 1055: 0.0062 - accuracy: 0.9991 - val_055: 0.3085 - val_accuracy: 0.9376 66/66 1-225 336ms/step - 1055: 0.0119 - accuracy: 0.9992 - val_1055: 0.34227 - val_accuracy: 0.9376 66/66 1-225 336ms/step - 1055: 0.0126 - accuracy: 0.9992 - val_1055: 0.3422 - val_accuracy: 0.9574 66/66 1-225 353ms/step - 1055: 0.0113 - accuracy: 0.9943 - val_acs: 0.3129 - val_accuracy: 0.9555 66/66 1-235 353ms/step - 1055: 0.0199 - accuracy: 0.9919 - val_acs: 0.2775 - val_accuracy: 0.9588 66/66 1-245 357ms/step - 1055: 0.0199 - accuracy: 0.9911 - val_1055: 0.2777 - val_accuracy: 0.9598 | Hon C File Coc | <pre>summy summy summy</pre> | set, validation | Connect | 2. Elass_ | → Share © Col weight= | c P ab Al Vone, | , " |
| Epoch 1 215 353m3/step 10555 00066 700712 10271 10271 10271 000512 0 | Hon C File Coc | <pre>summy summy summy</pre> | set, validation | Comment Connect | 2. Class_ | - Share ⊕ Col | c P ab Al Vone, | , " |
| Epoch 2/100 1 245 361ms/step - loss: 0.0100 - val_accuracy: 0.9943 - val_accuracy: 0.9521 Epoch 80/180 - 235 348ms/step - loss: 0.0062 - accuracy: 0.9943 - val_acs: 0.318 - val_accuracy: 0.9521 Epoch 80/180 - 235 348ms/step - loss: 0.0062 - accuracy: 0.9941 - val_acs: 0.3085 - val_accuracy: 0.9599 66/66 - - 225 336ms/step - loss: 0.0119 - accuracy: 0.9962 - val_acs: 0.3085 - val_accuracy: 0.9376 66/66 - - 245 363ms/step - loss: 0.0126 - accuracy: 0.9962 - val_acs: 0.3129 - val_accuracy: 0.9543 66/66 - - 235 353ms/step - loss: 0.0129 - accuracy: 0.9555 66/66 - - 245 357ms/step - loss: 0.0199 - accuracy: 0.9518 66/66 - - 245 357ms/ | Hon C File Coc | <pre>summy summy summy</pre> | set, validation | Comment Connect | 2024 * | - ☆ Share | c P ab Al Vone, | |
| Epoch 80/180 235 340ms/step - loss: 0.0062 - accuracy: 0.9981 - val_oss: 0.3085 - val_accuracy: 0.9599 66/66 Fench 81/180 - 235 340ms/step - loss: 0.0062 - accuracy: 0.9981 - val_loss: 0.3085 - val_accuracy: 0.9376 66/66 Fench 82/180 - 245 336ms/step - loss: 0.0119 - accuracy: 0.9982 - val_loss: 0.3462 - val_accuracy: 0.9543 66/66 Fench 83/180 - 245 363ms/step - loss: 0.0113 - accuracy: 0.9943 - val_loss: 0.3129 - val_accuracy: 0.9543 66/66 Fench 83/180 - 235 335ms/step - loss: 0.0119 - accuracy: 0.99962 - val_loss: 0.3129 - val_accuracy: 0.9555 66/66 Fench 83/180 - 245 357ms/step - loss: 0.0199 - accuracy: 0.9919 - val_loss: 0.3277 - val_accuracy: 0.9588 66/66 Fench 83/180 - 245 357ms/step - loss: 0.0199 - accuracy: 0.9919 - val_loss: 0.2775 - val_accuracy: 0.9588 66/66 Fench 83/180 - 235 335ms/step - loss: 0.0093 - accuracy: 0.9971 - val_loss: 0.2777 - val_accuracy: 0.9598 Epoch 86/100 - 235 335ms/step - loss: 0.0093 - accuracy: 0.9971 - val_loss: 0.2777 - val_accuracy: 0.9599 | Hon C File Coc | summy X OP brain_tumor.lpynb - Colab X + C Is colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Drain_tumor.ipynb X le Edit View Insert Runtime Tools Help Allchanges.saved ode + Text Found 2102_images belonging to 2 classes. Found 808_images belonging to 2 classes. Point_time_relevance > history = classifier.fit_generator(training_set, steps_per_epoch=None, epochs=100, verbose=1, callbacks=None, validation_data=test_ > 66/66 [================================ | set, validation | Comment Connect | Elass_ | - Share | ab Al | . · |
| Epoch 81/100 225 336ms/step - loss: 0.0119 - accuracy: 0.9962 - val_coss: 0.4227 - val_accuracy: 0.9376 Epoch 82/100 245 356ms/step - loss: 0.0126 - accuracy: 0.9962 - val_coss: 0.4227 - val_accuracy: 0.9543 Epoch 82/100 245 353ms/step - loss: 0.0126 - accuracy: 0.9962 - val_coss: 0.4227 - val_accuracy: 0.9543 Epoch 83/100 235 353ms/step - loss: 0.0113 - accuracy: 0.9962 - val_coss: 0.3129 - val_accuracy: 0.9555 Epoch 84/100 245 357ms/step - loss: 0.0199 - accuracy: 0.9919 - val_coss: 0.2775 - val_accuracy: 0.9588 Epoch 85/100 265 353ms/step - loss: 0.0093 - accuracy: 0.9971 - val_loss: 0.2777 - val_accuracy: 0.9599 Epoch 86/100 235 353ms/step - loss: 0.0093 - accuracy: 0.9971 - val_coss: 0.2777 - val_accuracy: 0.9598 | Hon C Coc Coc | summy X OP brain_tumor.lpynb - Colab X + C 1; colab/research.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Drain_tumor.ipynb 1 le Edit View Insert Runtime Tools Help Allchanges.saved odde + Text Found 2002_images belonging to 2 classes. Found 808 images belonging to 2 classes. Pound 808 images belonging to 2 classes.) history = classifier.fit_generator(training_set, steps_per_epoch=None, epochs=100, verbose=1, callbacks=None, validation_data=test_) 66/66 [================================ | set, validation | Comment Connect | 2. Class_ | | c P ab Al | |
| Epoch 82/180 1 245 363ms/step 10ss: 0.126 accuracy: 0.9943 val_oss: 0.3462 val_accuracy: 0.9543 66/66 [iman=100] 23s 353ms/step 10ss: 0.0126 accuracy: 0.9943 val_oss: 0.3129 val_accuracy: 0.9555 66/66 [iman=100] 24s 357ms/step 10ss: 0.0199 accuracy: 0.9919 val_oss: 0.2775 val_accuracy: 0.9588 66/66 [iman=100] 24s 357ms/step 10ss: 0.0199 accuracy: 0.9919 val_oss: 0.2775 val_accuracy: 0.9588 66/66 [iman=100] 24s 357ms/step 10ss: 0.0199 accuracy: 0.9919 val_oss: 0.2775 val_accuracy: 0.9588 Epoch 85/100 [iman=10] 23s 353ms/step 10ss: 0.0093 accuracy: 0.9971 val_oss: 0.2717 val_accuracy: 0.9599 Epoch 86/100 [iman=10] 23s 353ms/step 10ss: 0.0093 accuracy: 0.9971 val_oss: 0.2717 val_accuracy: 0.9599 | Hom C File Coc | summy X OP brain_tumor.lpynb - Colab X + C 1; colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Drain_tumor.ipynb 1 le Edit View Insert Runnime Tools Help Allchanges.saved odde + Text Found 2002_images belonging to 2 classes. Found 808 images belonging to 2 classes. Point_time_relevance of/d6 [==================================== | set, validation | Comment Connect | and a second sec | | c The second sec | , " |
| Epoch 82/100 23s 353ms/step Loss: 0.0113 accuracy: 0.9962 val_accuracy: 0.3129 val_accuracy: 0.9555 Epoch 84/100 - 24s 357ms/step loss: 0.0199 accuracy: 0.9919 val_acs: 0.3129 val_accuracy: 0.9555 Epoch 85/100 - 24s 357ms/step loss: 0.0199 accuracy: 0.9919 val_loss: 0.2775 val_accuracy: 0.9588 Epoch 85/100 - 23s 353ms/step loss: 0.0093 accuracy: 0.9971 val_loss: 0.2717 val_accuracy: 0.9599 Epoch 86/100 - 23s 353ms/step loss: 0.0093 accuracy: 0.99911 val_loss: 0.2717 val_accuracy: 0.9599 | Hon C Coc Coc | summy X OP brain_tumor.lpynb - Colab X + C Is colabresearch.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa Drain_tumor.ipynb X le Edit View Insert Runnime Tools Help Allchanges.saved odde + Text Found 2002_images belonging to 2 classes. Found 808 images belonging to 2 classes. history = classifier.fit_generator(training_set, steps_per_epoch=None, epochs=100, verbose=1, callbacks=None, validation_data=test_ 06/66 [================================== | set, validation | Comment Connect | 2. Class_ | - Share ⊕ Col | c P ab Al | |
| Epoch 84/100 66/66 [========] - 245 357ms/step - loss: 0.0199 - accuracy: 0.9919 - val_loss: 0.2775 - val_accuracy: 0.9588 66/66 [========] - 235 353ms/step - loss: 0.0093 - accuracy: 0.9971 - val_loss: 0.2717 - val_accuracy: 0.9599 Epoch 86/100 | Hon C Coc | Summy X OP brain_tumor.lpynb - Colab X + C 1: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Drain_tumor.ipynb 1: le Edit View Insert Rumine Tools Help Allchanges.saved odde + Text Found 2102_images belonging to 2 classes. Found 808_images belonging to 2 classes. Found 808_images belonging to 2 classes. Point 2102_images belonging to 2 classes. Point | set, validation | Comment Connect | 2. Class_ | - Share ⊕ Col | c D ab Al None, | |
| Epoch 85/100 66/66 [========] - 23s 353ms/step - loss: 0.0093 - accuracy: 0.9971 - val_loss: 0.2717 - val_accuracy: 0.9599 Epoch 86/100 | File Coco | Summy X OP brain_tumor.lpynb - Colab X + C 1: colab/research.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa D brain_tumor.ipynb 1: le Edit View Insert Runnime Tools Help Allchanges.saved odde + Text Found 2002_images belonging to 2 classes. Found 898 images belonging to 2 classes. Point 2102_images belonging to 2 classes. Point 2104_images belonging to 2 | set, validation | Comment Connect | Lass_ | - Share ⊕ Col | c P ab Al Vone, | , " |
| Epoch 86/100 | Hom C File Coc | Summy X OP brain_tumor.lpynb - Colab X + C Is colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Derain_tumor.ipynb Image: State Colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Derain_tumor.ipynb Image: State Colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Derain_tumor.ipynb Image: State Colaburesearch.google.com/drive/13ApQ9m-bsEbxOToTOHv0gVX_KuVbSbGa Image: State Colab | set, validation | Connect | 2.lass_ | - Go Col | c Definition of the second sec | , m |
| | Hon C File Cox | <pre>smm; smm; smm; smm; smm; smm; smm; smm;</pre> | set, validation | Connect | 2. Class_ | - G Col | c ab Al | |



www.ijasem.org

Vol 18, Issue 1, 2024

| Home - Google Drive X CO brain, tumor.ipynb - Colab X + | - 0 X |
|---|-------------------------------------|
| ← → C 🔓 colab.research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gVX_KuVbSbGa | 🖈 🕢 Error 🗄 |
| CO ⁶ brain_tumor.ipynb File Edit View Insert Runtime Tools Help <u>All changes saved</u> | 📮 Comment 😩 Share 🌣 👘 |
| = + Code + Text | Connect 🔻 🕲 Colab Al 🔥 |
| <pre>plt.title('model accuracy') plt.ylabel('accuracy') plt.xlabel('accuracy') plt.klabed('epoch') plt.legend('train', 'test'], loc='upper left') plt.show()</pre> | |
| Ca (2) model accuracy | |
| 100 905 900 905 900 905 900 905 900 905 900 905 900 905 900 905 900 905 900 905 900 905 905 | |
| <pre>[] # 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()</pre> | |
| | • X |
| 927F 🚆 Q Search 💔 🖬 🗭 🚫 🚍 🖸 🔯 🧮 🦉 | ヘ ^{ENG} (家 の) 22-04-2024 Q |





www.ijasem.org

Vol 18, Issue 1, 2024

| A Home - Google Drive X CO brain_tumor.lpynb - Colab X + | - o × |
|--|--|
| ← → C colab.research.google.com/drive/13ApQ9m-bsEbxOToTOhr0gV | KuVbSbGa 🔄 👘 Eror : |
| CO Co File Edit View Insert Runtime Tools Help <u>All changes saved</u> | 📮 Comment 🔹 Share 💠 👘 |
| HE + Code + Text | Connect 👻 🎯 Colab Al 🖍 |
| Q (2) | |
| <pre>{x} {] image1 = cv2.imread('/content/drive/MyDrive/brain_new_dat img = cv2.cvtColor(image1, cv2.c000B_BG82GAAY) cvt_thresh1 = cv2.threshold(img, 120, 255, cv2.THRESH_B1 cv2_imshou(img) cv2_imshou(img) cv2_imshou(thresh1) cv2_imshou(thresh2) cv2_imshou(thresh2) cv2_imshou(thresh3)</pre> | et/test/glioma_tumor/y1101.jpg') RY) RY_INY) C) |
| | |
| 92"F | |
| Sunny Q Search | |

Conculsion:

Without pre-trained Keras model, the train accuracy is 97.5% and validationaccuracy 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..



www.ijasem.org

Vol 18, Issue 1, 2024

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. Gonalves, 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 braintumor 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.



www.ijasem.org

Vol 18, Issue 1, 2024

[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.