GPU based Segmentation and Classification of Brain Tumour from MRI Images

Joel T. Nanthikattu, Navaneeth N, Gokul D, Sonette T, Binesh T, Binish M C, Vinu Thomas

Government Model Engineering College, Thrikkakara, Kochi, Kerala, India

Corresponding author: Joel T. Nanthikattu, Email: joelnanthikattu@gmail.com

Detection of Brain Tumour in its initial stage leads to an advancement in treatment methods. This paves the way for a consequent higher rate of life expectancy for the patient. These tumours can be successfully assessed by Magnetic resonance imaging (MRI). Alternatively, the immense amount of data generated by MRI causes manual segmentation to be highly time-consuming, thereby restricting the use of precise quantitative measurements in clinical practice. Brain tumours possess sizeable spatial and structural inequality among them which makes automatic segmentation a demanding task. Here in this work, Convolutional Neural Networks (CNN) are utilized to devise an accurate and efficient real time tumor tracking algorithm for detecting the different types of brain tumours. Python programming language is used for the development and the implementation is carried out on a typical Graphics Processing Unit (GPU).

Keywords: Magnetic Resonance Imaging, Machine learning, Graphic Processing Unit, Segmentation.

2021. In Rahul Srivastava & Aditya Kumar Singh Pundir (eds.), *New Frontiers in Communication and Intelligent Systems*, 17–23. Computing & Intelligent Systems, SCRS, India. https://doi.org/10.52458/978-81-95502-00-4-3

1 Introduction

Various non-invasive techniques of examining the internal regions of the human body are featured in the field of Medical Imaging. The images so obtained are different in their modalities and need to be processed for the diagnosis and therapy of diseases and disorders in humans. Successful identification and treatment of diseases can be attained in the most decisive and detrimental step of Image segmentation [1]. The primary aim of this step is effective computer vision, making tumour or lesion detection highly accurate. In recent times, on observing the statistics of death rate due to brain tumours which are alarming, huge efforts were directed to the extensive study of images of this particular type of lesions [2]. Three types of most pronounced brain tumours are Glioma, Meningioma, and Pituitary tumour. The chance for survival and increase in life expectancy of the patient is significantly enhanced by the detection of brain tumour in its initial stage and subsequent therapy. Magnetic Resonance Imaging is the most effectual practice for brain tumour detection. The fact that a gigantic volume of data is involved is what makes tumour segmentation from these MRI images a most prolonging and challenging task [3]. Moreover, the tumours being very ill-defined at their soft tissue boundaries, it is an extremely tedious work to do their segmentation accurately. Manual segmentation involves massive efforts, while automatic segmentation is also highly strenuous due to large variability of spatial structures involved [4]. The aim of this work is to develop an efficient algorithm to detect the three types of brain tumours and to implement this as a parallel processing task on a GPU. Image analysis in a quantitative manner is performed using CNN based machine learning with multiple iterations. The code for the CNN model was developed using Python. Training of the model was done using MRI dataset obtained from the MICCAI BraTs website [5]. This work attempts to present an accurate and efficient prediction of different types of tumours and the extraction of their features on the Nvidia Jetson Nano GPU. The experimental results were thoroughly compared to various models in terms of accuracy and sensitivity.

2 Literature Review

The analysis can be made more deliberate by dividing the specific regions of interest of an image in the particular step of Image segmentation. Upon the attainment of the appropriate level of subdivision, the segmentation should end [3]. MRI is a safer and better noninvasive modality which indicates the true dimensions of the brain, on comparison with other diagnostic methods [6]. For the semi-automatic and automatic segmentation and detection of brain tumours, the methods fall into two categories: intelligent and non-intelligent [7]. The schemes used to implement major intelligent type segmentation systems include Artificial Neural Networks, Support Vector Machines, Particle Swarm Optimization and Fuzzy C-means [3, 4]. Deep learning or Machine learning techniques based CNNs have emerged to be high-tech performers for brain tumour MRI image segmentation and classification [8-9]. Several algorithms for image classification based on varied approaches are available in the literature [6, 10-12]. The BRaTs database [5] has been prominently deployed in most of these works. [4] Proposes an automatic segmentation method based on CNN, dealing with 3x3 kernels. [10] Describes a scheme for classification of brain tumour in MRI data. GPUs have been explored in many works as they are hugely capable of the segmentation and classification of MRI brain images [13-15] and have been shown to be several times faster than[16] conventional model processors. The effectiveness and capabilities of Machine Learning approach deployed in an Nvidia Jetson Nano platform with a dew computing approach to video image recognition have been proposed in [17] which led to the chief motivating idea behind this work.

3 Methodology

The three main stages of this proposed work can be listed as: Pre-processing, Classification using CNN and Post-processing. The block diagram representation of various steps involved is shown in Fig.1.

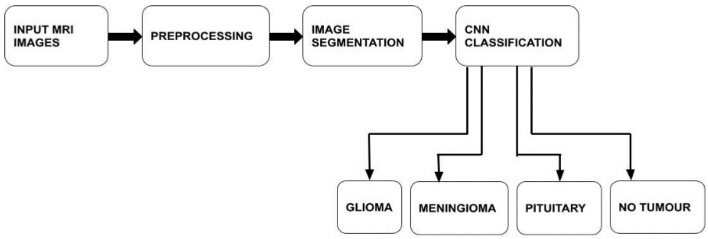

Fig. 1. Block diagram representation of the steps involved.

3.1 Pre-processing

The workflow begins with the initial phase of Image acquisition which is realized by inputting the BraTs dataset of brain tumour images taken from the medical database. Four categories of images comprise the dataset namely glioma, meningioma, pituitary tumour and the no tumour category. The dataset must be shuffled thoroughly in the next step to guarantee variance reduction. This shuffling step also helps to make the training reach its convergence faster, improves the model quality of machine learning, checks the model from learning the order of the training, and averts any bias from occurring during the training. The third step consists of data augmentation where different versions of a real dataset are generated artificially not only to expand its size but to enhance accuracy at the same time. This augmentation phase also includes noise addition, flipping, cropping, rotating and scaling. The unwanted distortions are eliminated and the important features are enhanced in the next step of pre-processing of the acquired images. The removal of bias-field distortions is achieved by the Batch normalization method [18] in this work. Additional layers are attached to a deep neural network so as to make them work swifter and more stable by this method. The introduction of internal normalization of the input values within the layer hastens the training phase of deep neural networks. A collated set of data at a time, known as a batch is used to carry out the training. Appropriate generalization by the model is also partly warranted by this batch normalization step. The balance in the value scales is ensured and proportionality in the range of values despite the scale change is also achieved. The mean intensity value and standard deviation among all training patches isolated out after each sequence computed subsequent to the normalization stage. Then, the normalization of the patches on each sequence is performed to ensure zero mean and unit variance.

3.2 Classification using CNN

Feature maps are obtained by the convolution of a signal or image with kernels which forms an important application of CNN. The weights of the kernels connect a unit in a feature map to the previous layer. The weights of the kernels are adapted during the training phase using the Back Propagation process. The non-linear RelU activation function is used in this work. The input volume is transformed into an output volume through a differentiable function by the stack of distinct layers in the CNN architecture. Convolutional layer, pooling layer and Dropout layer are the three individual layers that are commonly used. The core building block of a CNN is convolutional layer while the function of the pooling layer is nonlinear down sampling and drop out layer avoids over fitting. The two major phases of CNN based brain tumor classification are training and testing. The images are categorized under the four different labels of glioma, meningioma, pituitary and no tumor brain image. Pre-processing, feature exaction and classification with loss function are the steps performed in the training phase in order to make a prediction model. In the final step, automatic brain tumour classification is carried out using the convolutional neural network. ImageGenerator algorithm augments the dataset followed by Adam optimizer [19] training, with a mini-batch size of 16 and data being shuffled in every successive iteration. The dataset is split in the ratio of 80:20 ratios for training by CNN. Since this is a multi-classification problem the loss function employed is categorical cross entropy. The prediction process is carried out to get the four output classes mentioned earlier are obtained after the completion of the prediction process. Table 1 summarizes the designed CNN model.

3.3 Post-processing

Clusters that are below the level of a predetermined threshold are removed by employing volumetric constraints which also eliminates the possibility of small clusters being incorrectly classified as tumour. In the pre-surgical evaluation of non-lesionalepilepsy MRI post-processing should be included as it can provide principal values to improve the yield of structural MRI. Accurate identification and outlining of cortical abnormalities is guaranteed by this stage allowing them to be targeted and mapped more positively.

4 Hardware Description

The hardware implementation of the work was carried out using the NVIDIA Jetson-Nano Developer Kit as the GPU. Multiple neural networks can be run in parallel on this small yet powerful computer. Linux OS runs Jetson Nano with a performance of 472 GFLOPS of FP16 and power consumption of 5-10W. The specifications of the processor are shown in Table2.

5 Software Description

CUDA (Compute Unified Device Architecture) is a parallel computing platform and application programming interface (API) model devised to work with programming languages such as C, C++ and python. It uses a CUDA-enabled graphics processing unit (GPU) for general purpose processing. To streamline the GPU-based accelerated processing, a driver and runtime API for existing toolkits and libraries is provided by GPU-accelerated Computing with CUDA Python. The compilation of the complete Python program was performed on The Jupyter notebook environment. The Tensor Flow platform- run Keras API was used for the program.

6 Results and Discussion

The experiment is carried out on the BraTs dataset.3250 images are regenerated after the augmentation of the dataset images. Further division into training and testing groups of the total dataset in the ratio 80:20 is then performed. Randomly images were chosen from the dataset and segmentation was carried out. For a typical case with tumor is the segmented image is shown in Fig.2.Training of the model is performed on the training images. Two different graphs were plotted which depict the training and validation accuracy and loss versus the number of training epochs. Fig.3 illustrates final predicted result obtained after classification for an image section set. Respectively, 91% and 35% accuracy and loss were observed. Further validation of the results was implemented by computing the Confusion Matrix the accuracy and loss curves were plotted. These results are shown in Fig.4.

Table 1. CNN layers and their Properties

Layer No	Layer No	Layer No
1	Image input	256 x 256 x 3 images
2	Conv2D	256 x 256 x 64 with kernel size (3,3)
3	MaxPool2D	127 x 127 x 64 max pooling with stride (2,2)
4	Dropout	Dropout rate 0.2
5	Conv2D	125 x 125 x128 with kernel size (3,3)
6	MaxPool2D	62 x 62 x 128 max pooling with stride (2,2)
7	Dropout	Dropout rate 0.2
8	Conv2D	60 x 60 x 128 with kernel size (3,3)
9	MaxPool2D	30 x 30 x 128 max pooling with stride (2,2)
10	Dropout	Dropout rate 0.2
11	Conv2D	28 x 28 x 128 with kernel size (3,3)
1 2	MaxPool2D	14 x 14 x 128 with stride (2,2)
13	Dropout	Dropout rate 0.2
14	Flatten	Output shape 256
15	Dense	Output shape 16
16	Dropout	Dropout rate 0.25
17	Softmax	Softmax Classifier
18	Classification Output	Meningioma,Glioma,Pituitary and No tumor

Table 2. Jetson Nano Specifications

Component	Specification
CPU	Quad-core ARM A57 @ 1.43 GHz
GPU	128-core Maxwell
Video Encoder5	4K @ 30 — 4x 1080p @ 30 — 9x 720p @ 30 (H.264/H.265)
Storage	microSD
Memory	4 GB 64-bit LPDDR4 25.6 GB/s
Video Decoder	4K @ 60 — 2x 4K @ 30 — 8x 1080p @ 30 — 18x 720p @ 30 (H.264/H.265)
Connectivity	Gigabit Ethernet, M.2 Key E
Camera	2x MIPI CSI-2 DPHY lanes
Display	HDMI and display port
Mechanical	69 mm x 45 mm, 260-pin edge connector
USB	4x USB 3.0, USB 2.0 Micro-B

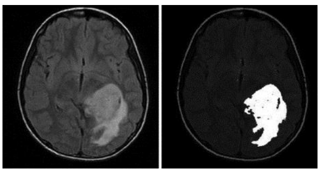

Fig.2. Segmented image with tumour

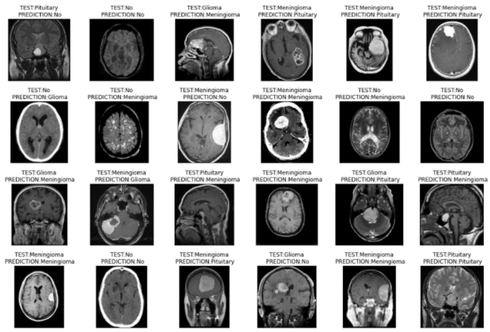

Fig.3. Predicted images

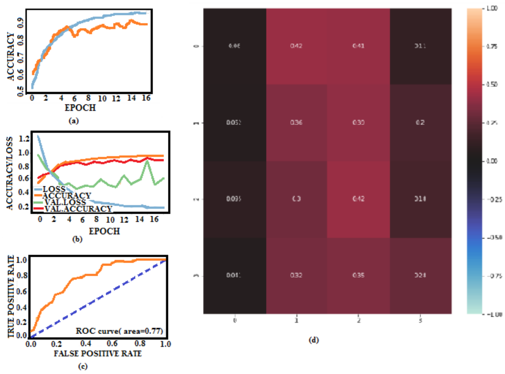

Fig.4. a) Accuracy b) Validation accuracy and loss c) Receiver Operating Characteristic (ROC) d) Confusion Matrix

7 Conclusion

Brain tumor segmentation and classification has been achieved in this work by employing MRI images. Classification using traditional classifiers and CNN follows the segmentation process. For better results, we implemented CNN with a split ratio of 80:20 of 3250 images, i.e. 80% of training images and 20% of testing images which brought in the accuracy 90%. The observation is that parallel processing aids in faster and efficient detection of brain tumors from MRI images. GPU computing is also found to be faster and more efficient. It is essential to parallelize and utilize high-performance computing platform in the context of the full dataset for maximum efficiency. Here the work was done using 2D Images only while it can be further extended to include 3D brain images also for improved results. To get a more accurate model for brain tumor the number of images in the dataset used can be further increased. In view of the future scope the work can also be extended to identify tumors that occur in other parts of the body.

References

[1] M. C. Clark et al., “Automatic tumor segmentation using knowledge based techniques”, *IEEE Trans. on Medical Imaging*, vol. 17, no. 2, pp. 238-251, 1998.

[2] T. Logeswari and M. Karnan, “An improved implementation of brain tumor detection using segmentation based on soft computing”, *J. Cancer Res. Exper. Oncology*, vol. 2, no. 1, pp. 6-14, 2010.

[3] T. Hossain et al., “Brain Tumor Detection Using Convolutional Neural Network”, in the *1st Int. Conf. Adv. Sci., Eng. Robotics Tech.,* 2019, pp. 1-6

[4] S. Pereira et al., “Brain Tumor Segmentation Using Convolutional Neural Networks in MRI Images”, *IEEE Tran. Med. Imag.*, vol. 35, no. 5, pp. 1240-1251, 2016.

[5] B. Menze et al., “The multimodal brain tumor image segmentation benchmark (BRATS),” *IEEE Trans. Med. Imag.*, vol. 34, no. 10, pp. 1993–2024, 2015.

[6] A. Mustaqeem, A. Javed and T. Fatima, “An Efficient Brain Tumor Detection Algorithm Using Watershed & Thresholding Based Segmentation”, *Int. J. Image, Graphics and Signal Processing*, vol. 10, pp. 34-39, 2012.

[7] M. Saii and Z. Kraitem, “Automatic Brain tumor detection in MRI using image processing techniques”, *Biomed. Stat. Inform.*, vol. 2, no. 2, pp.73-76, 2017.

[8] M. W. Nadeem et al.,“Brain Tumor Analysis Empowered with Deep Learning: A Review, Taxonomy, and Future Challenges”, *Brain Sci.*, vol. 10, no. 118, 2020.

[9] M. Sajjad et al., “Multi-Grade Brain Tumor Classification using Deep CNN with Extensive Data Augmentation”, *J. Comp. Sci.*, 30, pp. 174-182, 2018.

[10] E. F. Badran, E. G. Mahmoud and N. Hamdy, “An Algorithm for Detecting Brain Tumors in MRI Images",in *the 7th Int. Conf. Cloud Comput. Data Sci. Eng. - Confluence*, 2017.

[11] J. Amin et al., “Detection of Brain Tumor based on Features Fusion and Machine Learning”, *J. Ambient. Intell. Humaniz. Comput.*, pp.1-17, 2018.

[12] K. Usman and K. Rajpoot, “Brain tumor classification from multi-modality MRI using wavelets and machine Learning”, *Pattern Anal. Appl.*, vol. 20, pp. 871–881, 2017.

[13] H.Y. Tsai et al., “GPU-Accelerated Features Extraction from MR Images”, *IEEE Access*, vol. 5, 2017.

[14] A. Eklund et al., “Medical Image Processing on the GPU:Past, Present and Future”, *Medical Image Analysis, 2013*.

[15] J. Marko et al., “Evolution and Trends in GPU Computing”, *Proceedings of the 35th International Convention, MIPRO*, 2012.

[16] T. Kalaiselvi1, P. Sriramakrishnan and K. Somasundaram, “CUDA Model for Accelerating Brain Image Segmentation and Classification”, *Proc. Conf. Comp. Methods, Commun. Tech. Inform.*, 2012, pp.79-84.

[17] S. Valladares et al., “Performance Evaluation of the Nvidia Jetson Nano Through a Real-Time Machine Learning Application”, In book: Intelligent Human Systems Integration, 2021.

[18] V. Thakkar, S. Tewary and C. Chakraborty, “Batch Normalization in Convolutional Neural Networks — A comparative study with CIFAR-10 data,” in *5th Int. Conf. Emerging Appl. Inform. Tech.*, 2018, pp. 1-5.

[19] https://keras.io/api/optimizers/.