Study of Methods for the Classification of Gliomas

Oliver Otcasek
University of Rochester
ootcasek@u.rochester.edu

Abstract

The identification of gliomas is a problem that has a significant bearing on the lives of patients. Convolutional neural networks have been generally successful in classifying gliomas, but any error in classification can mean the life or death of a patient. The lack of publicly available data on the subject means that computational methods of classifying gliomas must be accurate with little data (n 500 samples). In this paper, I examine the use of several methods of image classification in an attempt to clarify which methods are the most successful and should be pursued in future work. I decided upon examining the use of a random forest image classifier, a three-dimensional convolutional neural network, and a 2d convolutional network based on finding the "best" tumor slices and training multiple networks for each dimension and combining the resulting predictions. Overall, the random forest classifier yielded an accuracy rating of 56.32%, the three-dimensional convolutional network yielded and accuracy of 78.51%, and the slice-based two-dimensional network yielded an accuracy rating of 98.53%. This project was implemented in Keras and Tensorflow using an augmented version of the BRATS 2015 dataset.

1. Introduction

The correct classification of brain tumors is an exceedingly important step in the treatment of brain cancer, and early detection is the key to a better prognosis. The importance of the task of classification of gliomas has been studied for many decades, and although the accuracy rates of convolutional neural networks for this problem have been impressive, it is imperative that the accuracy for classifying gliomas is increased, as any error can mean the life and livelihood of a patient. The purpose of this study is to identify the best method with regard to classifying gliomas out of a set of different image classifiers. In this paper, I use the BRATS 2015 dataset of brain tumors, [22][5][8][6][7] which is comprised of 3d MRI scans of the brain. The images in BRATS 2015 are separated into high-grade (HGG) and low-grade glioma (LGG) classes, where low-grade is stage one or stage two cancer and high-grade is stage three or stage four cancer. The dataset contains a dataset of 274 directories of training examples separated into 220 high-grade glioma directories and 54 low-grade glioma directories. Each of these directories contains five different types of 3d MRI images. These training examples are then split, and some section is used as testing examples, as BRATS 2015 does not provide tumor grades for its test set.

It was clear early on that this project would be hindered by limited data, so after the training and testing data were split, data augmentation such such as cropping, rotating, and filtering occurred to increase the size of the training dataset. I decided to test three methods of image classification in this study. First, I tested a random forest classifier to attempt to classify the glioma grades. The idea behind this choice was that the random forest classifier would use an explicit feature detector and may potentially find specific features of a glioma. Secondly, I chose to use a 3d convolutional neural network. The implementation is a rather straight-forward convolutional network built to take 3d image stacks as input. Finally, the most successful method was to use separate 2d convolutional neural networks that would take the strongest tumor slice from each axis of each 3d image stack. The "strongest" slice is the slice of a 3d segmented tumor image, which corresponds directly to some image slice in the 3d image stack of a brain.

2. Related Work

The issue of classifying gliomas has been studied before and continues to be studied often today. A recent paper by Ying Zhuge et al [27] proposes two separate methods to classify glioma grades in the same way that this paper does (classifying them as high or low-grade gliomas). Their work uses T1 and T2-weighted images from the BRATS 2018 dataset, which is of the same form as BRATS 2015, and TCIA low-grade glioma data. [14] [24] First, they perform inhomogeneity correction in their dataset, which corrects shading artifacts in MRI images. Both of their methods rely on a u-network neural network to segment the tumor from the 3d images. In their first method of classifi-

cation, they use data augmentation to increase the size of their training dataset. They then use an R-CNN neural network. Secondly, they use three-dimensional CNN, which is a three-dimensional volumetric CNN that is applied to only the region that makes up the segmented tumor. Note, their study is similar to mine, in that it is also seeking to find the best method to classify gliomas in the same way that my study does, and it also uses many of the same preprocessing techniques, which are standard practice for the implementation of CNNs for medical usage. They report the accuracy of their first method at 96.63%, and their second method at 97.1%. They trained their first method for a total of ten and a half hours, and their second method for three hours.

Another recent paper by Mzoughi et al. [23] achieved an accuracy of 96.49% using a three-dimensional CNN on T1 weighted MRI images, which are images that use short TR (repetition time; the amount of time between pulses for a slice) and short TE (time to echo; the time between the receiving of the RF signal and the receipt of the echo signal). [26] They classify the images into the same classes as I do and the aforementioned paper do; into HGG and LGG classes.

Badza et al. [4] used a new CNN architecture to classify gliomas into three different classes. They note that their method is simpler than many existing CNNs created to solve the same problem, yet the results of their experiments seem to prove that even simple networks work well on this problem. Using T1-weighted images from the Brain Tumor Dataset by Jun Cheng of the Tianjin Medical University in China, [12] their CNN classified brain tumors as meningioma, glioma, or pituitary. The images of the Brain Tumor Dataset they used come in three plains; the axial plane (birdseye view), the coronal plane (from the back of the head), and the sagittal view (from the side). Note that this is similar to how I split the three-dimensional slices into separate planes to test the idea of multiple two-dimensional CNNs working to classify a three-dimensional image. Badza et al were able to achieve an accuracy rating of 96.97%, evaluated using 10-fold validation.

A paper by Ozlem Akar et al. [2] explored the use of random forest image classification for the classification of multispectral satellite images. A random forest image classifier essentially works by building N different decision trees around a particular pixel, and each decision tree will return a separate classification of each pixel. Ozlem Akar et al. used this technique on satellite images, classifying them by land-use type, such as rural, Forest, Ocean, etc., and reported an accuracy rating between 77.92% and 87.08% on different images. This paper inspired me to try random forest classification on gliomas. I reasoned that perhaps the method could avoid some of the pitfalls of CNNs, such as texture bias (see Geirhos et al.).

3. Methods

For each different method mentioned in this paper, I use a NVIDIA T4 Tensor Core GPU and 25 GB of RAM, supplied by Google Collab Pro. Each method is coded in Python using Numpy 1.19 [17], Keras 2.4[13], Tensorflow 2.3[1], MedPy 0.1.0 [20], Sklearn .24 [25] [11], and Pandas 1.1.5 [21].

3.1. Preprocessing and Data Augmentation

Data preprocessing is an important step for medical imaging, given both the need for high accuracy and the lack of real world data open for general use. The data of the BRATS 2015 dataset is separated into 220 high-grade glioma (HGG) directories and 54 low-grade glioma (LGG) directories, each of which contain five mha image files that store T1, T2, T1c, and FLAIR weighted three-dimensional images, along with a mha image that contains a three-dimensional segmented image of the tumor. For the purposes of this study, I chose to use the T1 images due to several factors. First, the studies mentioned in the "Related Work" section use the T1 weighted images, which led me to use them as well so that my work is comparable to theirs'. Secondly, only one type of weighted MRI images should be used, as there are significant differences between the usage of different weights.

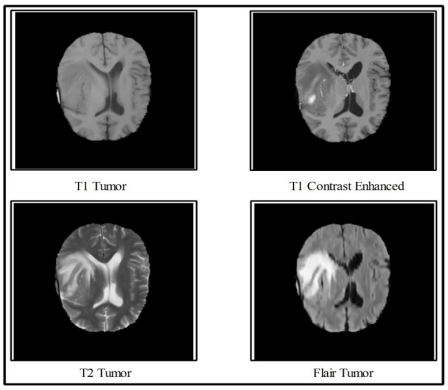


Fig. 1, Images from BRATS dataset; Khan M.A., Ashraf I., Alhaisoni M., Damaševičius R., Scherer R., Rehman A., Bukhari S.A.C. Multimodal brain tumor classification using deep learning and robust feature selection: A machine learning application for radiologists. Diagnostics. 2020;10:565. doi: 10.3390/diagnostics10080565. [19]

First, I converted each mha file to a dicom file, as dicom files have much greater support in medical imaging libraries in Python. Because mha and dicom files have the same structure, all that was needed was a simple script to change the filenames and modify the mha file's metadata (that includes important information on the patient and on the tumor.) Then, I normalized the images using the z-score normalization technique. The reason I chose z-score normalization is because it has been shown by Reinhold et al. to be an effective method for improving the accuracy of synthesizing images of brains when T1-weighted MRI scans

are used. It follows that z-score normalization should also be effective in improving the accuracy of classifiers using T1-weighted images.

Two of the methods I use (the random forest method and the two-dimensional slice method) use the strongest two-dimensional slices in each used T1-weighted and normalized dicom image. For this task, the tumor in the image needed to be segmented and its volumetric boundaries documented. Thankfully, the BRATS 2015 dataset provides the volumetric bounds of each tumor in the metadata of the mha files, which can be directly and safely ported to dicom files. So, I created a program that iterated through each slice along the x, y, and z axes to find the maximum slice in each axis. The tumor data within the dicom files are entirely white on a black background, so the maximum slice is the slice with the largest image of the tumor along the axis in which the three-dimensional image is being traversed. These strong slices were then saved in multiple directories pertaining to their respective axes. The slice files were also numbered appropriately to facilitate the segmentation of data into training and test sets. For example, the first dicom image in the BRATS 2015 dataset corresponds to patient one, and so all of the slice files will be named 1.png, and each directory for each axis will contain a file called 1.png which is the slice from patient one in the direction corresponding to the appropriate axis.

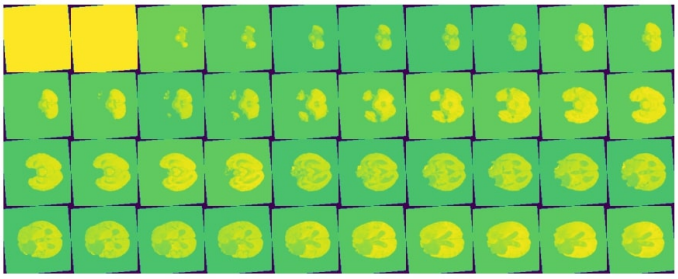


Fig 2: A montage of normalized, rotated (augmented) slices

All tissue has been stripped from every three-dimensional image (a technique called skull stripping.) BRATS 2015 images are stripped.

3.2. Method One: Random Forest

The first method I tried was a random forest image classifier. The use of random forest for tumor classification is not common. My reason for using this technique for this task is because random forest classifiers work on feature detection and extraction. I reasoned that perhaps this evaluation of many special features in a normalized slice may be able to find certain characteristics of a tumor that a CNN might miss. A paper by Geirhos et al. [15] concluded that some CNNs are particularly biased towards texture, but a random forest classifier should have no such bias. Since a random forest is simply an aggregation of decision trees, overfitting is limited. [9]

To begin a random forest classifier, the user must supply the constants M and N, M is used to signify the number of variables to split a node, and N is the number of decision trees. First, N samples are drawn from some fraction of a dataset, where the rest of the dataset is to be used as a test set. This test set is sometimes called "out-of-bag" data, which is used for the "out-of-bag" error calculation. Then, N trees are created and M "predictors" are chosen at random. Then, at each node, a split is performed according to the GINI index. When the GINI index is zero, the splitting halts and a class is predicted using all N trees. [3]

In my implementation, I use three different feature descriptors whose results are concatenated to create a final feature list. My choice of descriptors are meant to extract features based on texture and shape. For shape feature extraction, I used Hu moments. [18] For texture feature extraction, I used Haralick texture features. [16] I also used a histogram of oriented gradients (HOG) descriptor to simplify the image and improve accuracy.

In this project, I chose N = 50000 (50000 trees) and N = $\sqrt{M}$, where M is the cardinality of the set of all discovered features. The choice of the square root function is due to an analysis by Leo Breiman that states that the square root of the total number of features generally leads to nearly optimal results. [10]

3.3. Method Two: a Three-dimensional CNN

The second method, a three-dimensional CNN, is perhaps the most obvious method for classifying three-dimensional images. First, this classifier directly loads in the T1-weighted dicom images and uses z-score normalization on them. These images are labelled appropriately when they are loaded. Then, the training data is split for 10-fold cross validation.

The following CNN is simple, and I found this model to be the most accurate and suitable for demonstrating the power of a three-dimensional CNN for classifying gliomas. The CNN in question is shown in Fig 2. The input is a five-dimensional tensor with 64filters in the first two layers, 128 filters in the 3rd layer, and 256 filters in the 4th and final layer. Every activation function is a relu, except for the final activation function, which is a sigmoidal function.

3.4. Method Three: a Combination of Three Two-dimensional CNNs

The final method used in this study is a combination of two-dimensional CNNs. There are three CNNs, each of which takes a single slice that follows an axis along the x, y, or z direction. Once each CNN is trained, they each make a set of predictions about the test data, and the CNN with the highest accuracy rating is chosen from the other

two. Then, the output depends on which CNN has the highest accuracy at the end of training. The output of the main CNN may be the final prediction for the test set, but if the other two CNNs disagree on the classification of an image, and these other CNNs are accurate enough, then the result can be changed to increase the accuracy of the entire system. I have defined the following condition to determine if the other CNNs should be used:
$1 - accuracy_{chosen} > (1 - accuracy_{notchosen1}) * (1 - accuracy_{notchosen2})$ This condition I have defined is intended to state that the probability that the two unchosen CNNs are incorrect, if they agree on a classification, is less than the probability that the chosen CNN is wrong, if the chosen CNN and the two unchosen CNNs disagree on a classification.

So, for example, if the CNN in the x direction has an accuracy rating of 96% at the end of training, and the CNN in the y direction has an accuracy rating of 95%, and the CNN in the z direction has an accuracy rating of 94%, then the x-direction CNN would be chosen. Let the set M be the prediction of the x-direction CNN for the test data. Now, the condition $1 - .96 > (1 - .95) * (1 - .94)$. is true, so whenever the y-direction CNN and the z-direction CNN agree on a classification, but the x-direction CNN does not, the value in M will be the classification decided by the y and z-direction CNN.

4. Experiments

Each CNN was given two hours of training on their respective datasets. Because they were each given the same amount of time, their performance can be judged accurately. The random forest classifier was given a an N value of 100000 and an M value of 316 ($\sqrt{100000} = 316.2277...$).

4.1. Random Forest Method Results

Over twenty trials, the mean accuracy of the random forest method was 56.32%. For the precision and recall calculations, I used the HGG class as "positive" and LGG as "negative." The precision of this method was 60.5%, with a recall of 52.3%. It is evident from this that the classifier was far more likely to classify a HGG correctly than an LGG, which is not entirely surprising given that HGGs tend to be larger and more pronounced than LGGs, as they are a higher grade of cancer.

Unfortunately, the random forest method's accuracy was poor overall, even at 100000 trees. The poor accuracy was surprising, given the combination of features detected by both shape and texture. Overall, given these poor results, it seems unwise to pursue random forests as a method for the classification of gliomas, especially because even simple CNNs have shown much better results.

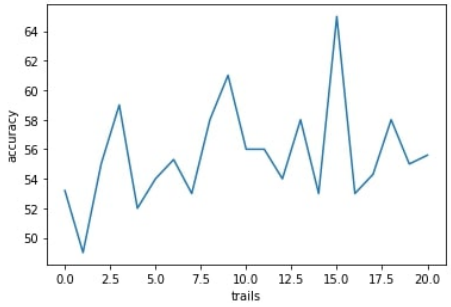


Fig 3: Accuracy of the random forest classifier over 20 trials

4.2. Three-dimensional CNN Method Results

The three-dimensional CNN was trained for three hours on the normalized, augmented set of T1-weighted images from the training data provided by the BRATS 2015 dataset. The final accuracy over the test data was 78.51% with a training set accuracy of 75.34%. The precision rate was 77.27% and the recall was 80.95%; meaning that the CNN was slightly more likely to classify HGGs correctly than LGGs. The f-score was 79.07%. See Fig. 10 for the CNN model used.

This method is clearly more accurate than the random forest method, yet there is still clearly a lot of room for improvement. The three-dimensional CNNs as described in the "related works" section tend to have an accuracy rating of 97 - 97%, which means that my implementation of a three-dimensional CNN for glioma classification is lacking. The results, however, do show that three-dimensional CNNs are effective in classifying gliomas, as the results are far better than random. One thing that is clear is that even a simple CNN like this vastly outperforms my implementation of a random forest classifier, which likely shows that CNNs are a much better option for the classification of gliomas.

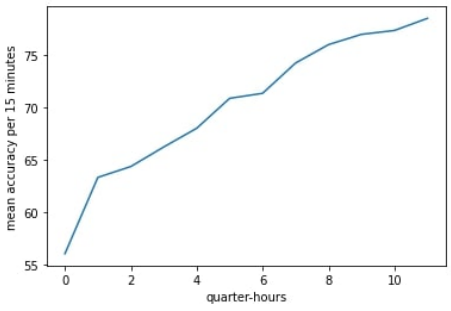


Fig 4: Mean accuracy of the 3d CNN every 15 minutes for 3 hours

4.3. Two-dimensional Combination of Networks Results

Each two-dimensional CNN was trained for three hours on specially chosen (the "strongest" tumor slices) modified

two-dimensional image slices taken from t1-weighted images from the BRATS 2015 dataset. After training, the y-direction CNN reported an accuracy rating of 97.6%, with a precision rate of 96.9% and a recall of 99.1%. The x-direction CNN reported an accuracy rating of 94.3%, with a precision rate of 92.23% and a recall of 95.63%. The z-direction CNN reported an accuracy rating of 96.53%, with a precision rate of 94.39% and a recall of 97.81%. Therefore, the y-direction CNN was chosen. See Fig. 11 for the CNN model used for all three CNNs.

Since $1 - .976 = .024$, and $(1 - .943) * (1 - .9653) = .0019779$, the final prediction sets were a composite of the results of the three CNNs. Whenever the y-direction CNN disagreed with both the x-direction and the z-direction CNN, then the result in the prediction set was changed to the result predicted by the x-direction and z-direction CNN. On ten trials of testing using random samples from the test set, the composite prediction set had an accuracy rate of 98.53%.

Clearly, the two-dimensional combination of networks produced the best results of the study, and even produced better results than the works mentioned in the "related works" section, even with a small dataset and a short training time. This method's promising accuracy results should be investigated further to see whether it has such promising results on three-dimensional images. After training the CNNs on the best slices, perhaps three-dimensional MRI scan slices can be fed into separate CNNs in the style of this method, so that it can classify three-dimensional scans without any sort of tumor extraction.

5. Conclusion

In conclusion, only the two-dimensional combination of networks produced promising results. The results of that method exceeded other state-of-the-art glioma classification CNNs proposed in recent years. The random forest method produced unsatisfactory results, as did the three-dimensional CNN. Given that other three-dimensional CNNs for glioma classification by other authors have shown better results, it is reasonable to imply that my implementation was substandard. However, the two-dimensional combination of networks produced very satisfactory results and should be studied further.

In all, this study was successful in identifying a promising candidate for future study. Since that was the goal of this project, I believe this project was successful.

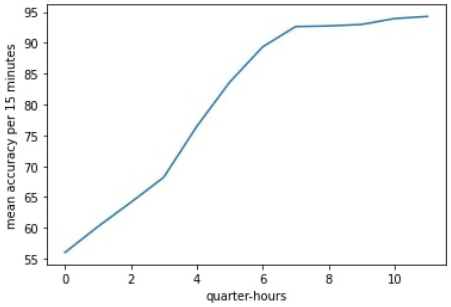


Fig 5: Mean accuracy of the x-direction CNN every 15 minutes for 3 hours

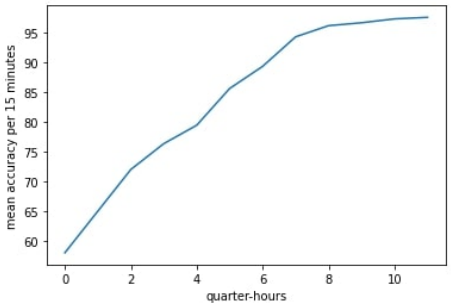


Fig 6: Mean accuracy of the y-direction CNN every 15 minutes for 3 hours

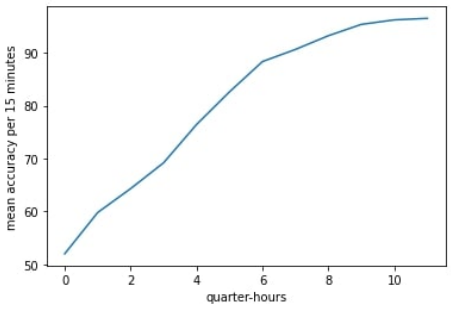


Fig 7: Mean accuracy of the z-direction CNN every 15 minutes for 3 hours

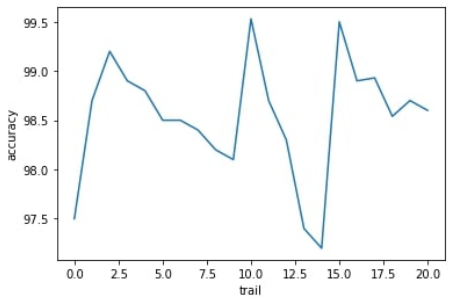


Fig 8: accuracy for 20 trails of 3 network composite

References

[1] M. Abadi and others year=2015. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. 2

[2] O. Akar and O. Gungor. Classification of multispectral im-

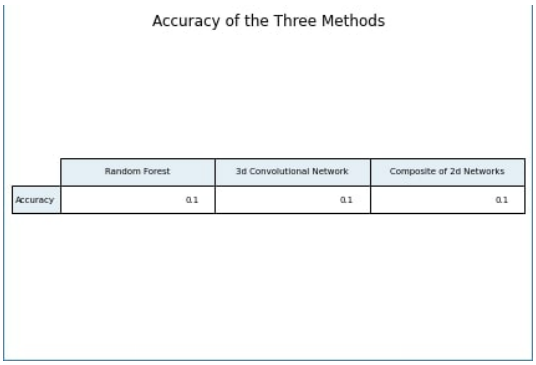


Fig 11: Accuracy of all Networks

ages using random forest algorithm. *UCTEAChamber of Surveying and Cadastre Engineers: Journal of Geodesy and Geoinformation*, 2012. 2

[3] O. Akar and O. Gungor. Classification of multispectral images using random forest algorithm. *UCTEAChamber of Surveying and Cadastre Engineers: Journal of Geodesy and Geoinformation*, pages 106–107, 2012. 3

[4] M. M. Badža and M. C. Barjaktarovic. Classification of brain tumors from mri images using a convolutional neural network. *MDPI AI*, 2020. 2

[5] S. Bakas, H. Akbari, A. Sotiras, M. Bilello, M. Rozycki, J. Kirby, J. Freymann, K. Farahani, and C. Davatzikos. Advancing the cancer genome atlas glioma mri collections with expert segmentation labels and radiomic features. sci data. *Sep 5;4:170117. doi: 10.1038/sdata.2017.117. PMID: 28872634; PMCID: PMC5685212.*, 2017. 1

[6] S. Bakas et al. Segmentation labels and radiomic features for the pre-operative scans of the tcga-gbm collection. *The Cancer Imaging Archive, DOI: 10.7937/K9/TCIA.2017.KLXWJJ1Q*, 2017. 1

[7] S. Bakas et al. Segmentation labels and radiomic features for the pre-operative scans of the tcga-lgg collection, 2017. 1

[8] S. Bakas et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. 2019. 1

[9] L. Breiman. Random forests. page 4, 2001. 3

[10] L. Breiman. Manual on setting up, using, and understanding random forests. 2002. 3

[11] L. Buitinck et al. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013. 2

[12] J. Cheng. Brain tumor dataset. 2017. Accessed: 2020-11-15. 2

[13] F. Chollet et al. Keras. https://keras.io, 2015. 2

[14] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, L. Tarbox, and F. Prior. The cancer imaging archive (tcia): Maintaining and operating a public information repository. *Journal of Digital Imaging, Volume 26, Number 6*, pages 1045—1057, 2013. 1

[15] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *ICLR 2019*, 2016. 3

[16] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 1973. 3

[17] C. R. Harris et al. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020. 2

[18] M. K. Hu. Visual pattern recognition by invariants. *IRE Trans. Info. Theory, vol. IT-8*, pages 179–187, 1962. 3

[19] M. Khan, I. Ashraf, M. Alhaisoni, R. Damaševičius, R. Scherer, A. Rehman, and S. Bukhari. Multimodal brain tumor classification using deep learning and robust feature selection: A machine learning application for radiologists. *Diagnostics. 2020;10:565. doi: 10.3390/diagnostics10080565.*, 2020. 2

[20] O. Maier et al. Medpy. In *Proceedings of the 9th Python in Science Conference*. Accessed: 2020-10-23. 2

[21] W. McKinney et al. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, volume 445, pages 51–56. Austin, TX, 2010. 2

[22] B. Menze et al. The multimodal brain tumor image segmentation benchmark (brats). *Oct;34(10):1993-2024. doi: 10.1109/TMI.2014.2377694. Epub 2014 Dec 4. PMID: 25494501; PMCID: PMC4833122*, 2015. 1

[23] H. Mzoughi, I. Njeh, A. Wali, M. B. Slima, A. BenHamida, C. Mhiri, and K. B. Mahfoudhe. Deep multi-scale 3d convolutional neural network (cnn) for mri gliomas brain tumor classification. *Journal of Digital Imaging*, 33:903—-915, 2020. 2

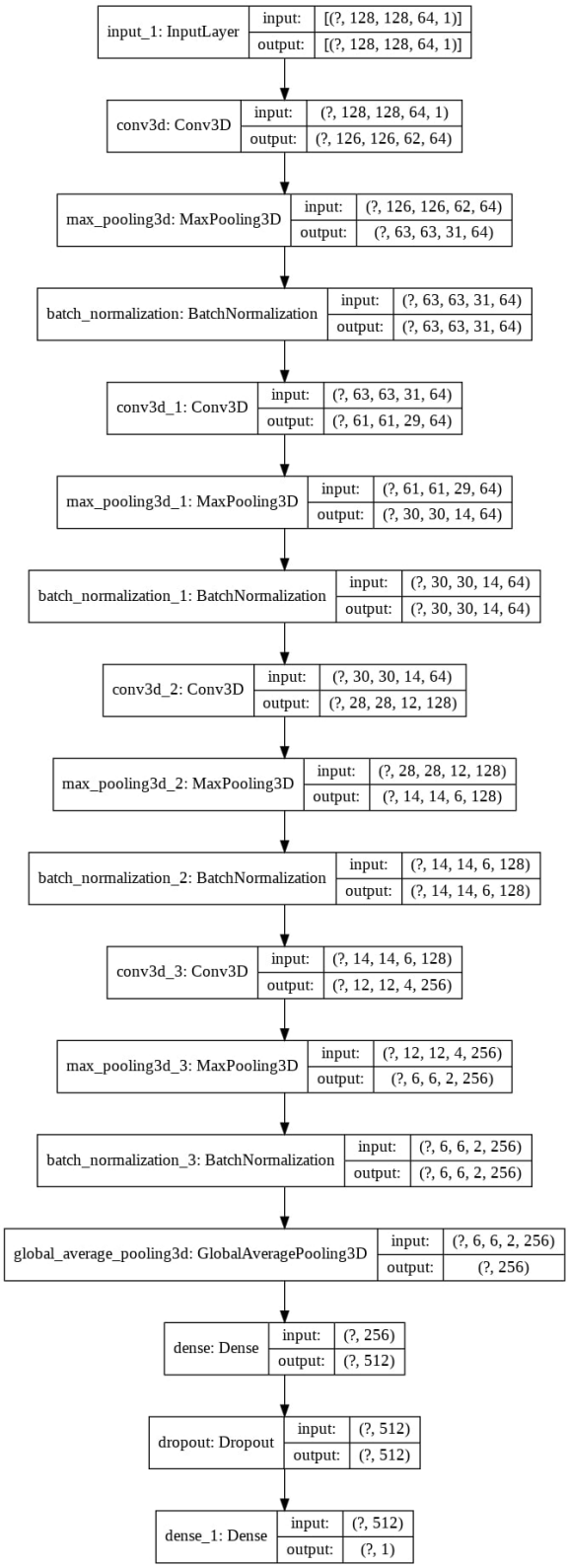
[24] N. Pedano, A. E. Flanders, L. Scarpace, T. Mikkelsen, J. M. Eschbacher, B. Hermes, and Q. Ostrom. Radiology data from the cancer genome atlas low grade glioma [tcga-lgg] collection. the cancer imaging archive. 2020. Accessed: 2020-11-15. 1

[25] F. Pedregosa et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 2

[26] D. C. Preston. Magnetic resonance imaging (mri) of the brain and spine: Basics. 2016. Accessed: 2020-10-23. 2

[27] Y. Zhuge, H. Ning, P. Mathen, J. Y. Cheng, A. V. Krauze, K. Camphausen, and R. W. Miller. Automated glioma grading on conventional mri images using deep convolutional neural networks. *Med. Phys., 47*, pages 3044–3053, 2020. Accessed: 2020-11-15. 1

6. Fig 9: Three-dimensional CNN



7. Fig 10: CNN Model for Method 3