A New Algorithm for Fully Automatic Brain Tumor Segmentation with 3-D Convolutional Neural Networks

Christopher Elamri
Stanford University
mcelamri@stanford.edu

Teun de Planque
Stanford University
teun@stanford.edu

Abstract

Glioblastoma (GBM) is the most common primary tumor of the central nervous system in the United States, with 3 in 100,000 people diagnosed every year [1]. In this paper, we present a new algorithm for fully automatic brain tumor segmentation based on 3-D Convolutional Neural Networks. Segmentation of the GBM region of the surrounding brain makes it easier to access the image data within MR scans, and consequently can help us better understand GBMs. Most methods for volumetric image data use 2-D convolutions; we present a true generalization of CNNs to 3-D filters to preserve spatial information and increase robustness. In addition, we present a relatively high bias CNN architecture that enables us to both expand the effective data size and reduce the variance of our model. Our median Dice score accuracy is around 89 percent in the whole tumor segmentation. This result represents a significant improvement over past algorithms and demonstrates the power of our approach in generalizing low-bias high-variance methods like CNNs to learn from medium-size data sets.

1. Introduction

Glioblastoma (GBM) is the most prevalent and most aggressive type of brain cancer [2]. As a result of its resistance to existing therapeutic methods, the most common length of survival following diagnosis is 12 to 15 months, and less than 3-5% of patients survive more than five years [1][3]. Much is still being done to understand GBM and its two subtypes, high grade glioma (HGG) and low grade glioma (LGG), but there is still a gold mine of untapped data. Chief amongst these is imaging data in the form of magnetic resonance (MR) scans. Most methods of analyzing and extracting quantitative information from this imaging data requires some form of segmentation of the GBM, and better yet, classification of the tumor into four sub-categories: necrosis, edema, non-enhancing tumor, and enhancing tumor. To date, the gold standard of segmentation is still human radiologist segmentation. However, with the pure size of imaging data being accrued, manual segmentation of all images is no longer a sustainable system.

Nonetheless, existing automated segmentation algorithms are based on high-bias learning algorithms and have failed to give substantial improvement in classification accuracy as described by Subbanna et al., Menze et al., and Bauer et al. [4]. As a result, these high-bias systems are increasingly being replaced with data-driven low-bias models. The development of low-bias systems such as convolutional neural networks for medical purposes is, nevertheless, challenging due to their high-variance, as many medical datasets are limited in size.

In this paper, we propose a novel approach to automatic brain tumor segmentation. We take as input a 3-D MR image and output one of five labels (normal or one of 4 tumor subtypes) for each voxel (the 3-D analog of pixel). Our new approach surpasses current methods by reducing the variance of CNNs applied to medium datasets. First, where most methods use 2-D convolutions for 3-D data, we introduce a CNN that uses 3-D filters, made computationally efficient by the transformation to Fourier space. Consequently, our system is more robust and minimizes loss of spatial information. Second, we present a novel CNN architecture which differs from those traditionally used in brain segmentation and computer vision [4][5]; our CNN architecture uses a higher-bias system than CNNs using pixel-wise neural networks that enables us to both expand the effective data size from patient number to total pixel and reduce the variance of CNNs. Our algorithm for GBM segmentation and characterization exploits the CNNs framework and the structure of medical imaging data to train a model that not only is highly robust but also generalizes data-driven low-bias systems to medium data while minimizing variance.

2. Problem Statement

Our problem amounts to efficiently classifying brain voxels into five categories: non-tumor, necrosis, edema, non-enhancing, and enhancing. The input to our system consists of a preprocessed 3-D image of a brain. Each 3-D image is of size 240 × 240 × 155 × 4, where the depth 4 represents the four modalities of our input images. The goal is to for each voxel correctly and efficiently output its category (non-tumor, necrosis, edema, non-enhancing, and enhancing).

2.1. Dataset

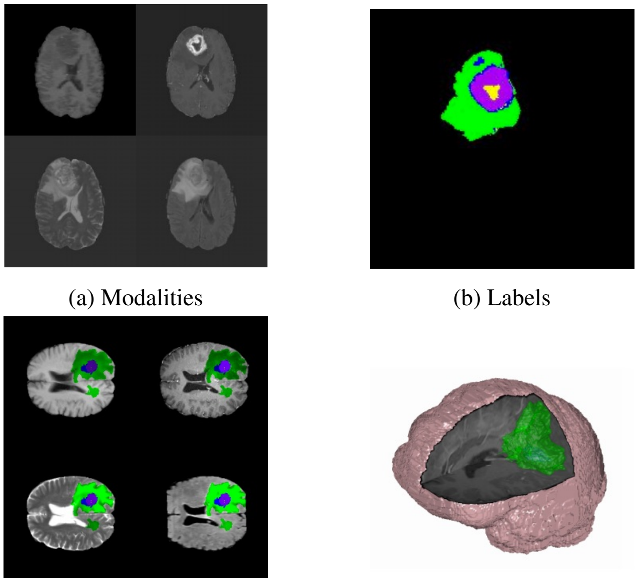
We use the data provided by the MICCAI Multimodal Brain Tumor Image Segmentation Challenge (BRATS) [4]. The 2015 BRATS dataset contains 274 patients: 220 patients with high grade glioblastomas and 54 patients with low grade glioblastomas. For each patient our data contains four MR modalities: T1 post-contrast, T1 pre-contrast, T2 weighted, and FLAIR. In addition, the data contains an expert segmentation for each patient that we will treat as the ground truth. The expert segmentation was made by pooling together the segmentation of eleven different radiologists and includes pixel-wide labeling as one of five categories: non-tumor, necrosis, edema, non-enhancing, enhancing.

All the data has been preprocessed, involving skull stripping, co-registration of the images, and interpolation of the images so that all images are of the same size (240 × 240 × 155 pixels). We were not given any information about the spatial scale of the pixels; and thus, all units for our current method will be in the given pixel-space.

3. Technical Approach

3.1. Convolutional Neural Network Motivation

Convolutional neural networks (CNNs) are powerful low-bias systems in deep learning with remarkable performance in complex tasks ranging from facial recognition to image classification [7][8]. CNNs use 2D convolutions as filters for feature learning and feed the results of these convolution layers into a fully connected neural network. Our algorithm builds on the idea of a CNN as a fully connected neural network trained on sequential convolution layers to improve accuracy and computational efficiency in learning from 3D imaging data. Specifically, we generalize the idea of 2D convolutions to 3D filters that retain the full spatial information of the original stack of images. We then decouple the pixels when providing the training data for a fully connected neural network, thus increasing the size of our training set and providing a computational speedup.



(a) Modalities

(b) Labels

(c) Modalities

(d) 3D-Rendering of Tumor

Figure 1: **Data Visualization.** (a) All four modalities are co-registered. The modalities are ul: T1-pre, ur: T1-post, dl: T2W, dr: FLAIR. (b) Corresponds to (a). Better visualization of the four subregions of the tumor (Y: necrosis, G: Edema, B: Non-Enhancing, P: Enhancing). (c) Visualization of the colored labels on top of the modality images for a different patient. (ul: T1-pre, ur: T1-post, dl: T2W, dr: FLAIR) (d) 3-D rendering of the tumor within the brain.

3.2. 3-D Convolutions

We can reduce the bias of our system by considering the input image as a 3-dimensional space of pixels. Expanding our convolution from our usual 2-dimensional convolutions into 3-dimensional convolutions, we get the following equation:

$$(I * f)[x, y, z] = \sum_{\tau_x=1}^{240} \sum_{\tau_y=1}^{240} \sum_{\tau_z=1}^{155} I[\tau_x, \tau_y, \tau_z] \dot{f}[x - \tau_x, y - \tau_y, z - \tau_z]$$

where I is the 3-dimensional image and f is the 3-dimensional filter. Given an image of size $n \times n \times n$ and a filter of size $m \times m \times m$ the time complexity of the convolution operation is $O(m^3 n^3)$. We can reduce the time complexity of the convolution operation to $O(n^3 log(n))$ by implementing our convolution as an element-wise multiplication in Fourier space. We use a 3-dimensional Difference of Gaussian filter:

$$DoG(\sigma) = \frac{1}{(2\pi\sigma^2)^{3/2}} \exp \frac{-x^2 + y^2 + z^2}{2\sigma^2} - \frac{1}{(\pi\sigma^2)^{3/2}} \exp \frac{-x^2 + y^2 + z^2}{\sigma^2}$$

Each filter is a difference of two 3-dimensional Gaussians (normalized) with scales $\sqrt{2}\sigma$ and σ. We create 8 of these filters with the scales $[\sqrt{2}, 2, 2\sqrt{2}, ..., 16]$. Difference of

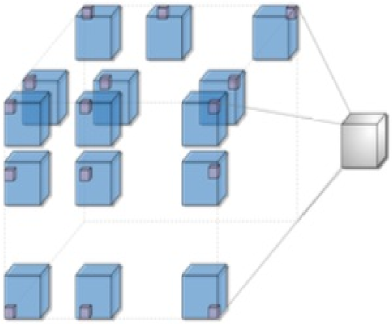


Figure 2: **Visualization of 3-D Convolution.** Similar to the 2-dimensional equivalent, in a 3-dimensional convolution, we create an output image from the dot-product of our 3-dimensional filter and a sub-image centered at every pixel. We represent the filter with the red-box, our image with blue box, and the convolutional product with our white box

Gaussian is an effective blob detector since it is rotationally symmetric.

To create our final convolutional products, we will consider all four modalities, and the magnitude of the gradient of those images. In addition to the eight filter products, we will include the original pixel intensity and the magnitude of the gradient value. Thus, for each of the four modalities, we will get an expansion into 18 feature images, giving us a total feature space of 72-dimensions for each of our pixels. We choose a filter of size 33 × 33 × 33. Basically, we get for each voxel $[x, y, z]$ the following set of features:

- 1 voxel intensity $V[x, y, z]$
- 1 voxel intensify gradient $\nabla V[x, y, z]$
- 8 DoG convolutions $(V * DoG)[x, y, z]$
- 8 DoG convolutions in gradient space $(\nabla V * DoG)[x, y, z]$

3.3. 3-D Convolutional Neural Network Architecture

Our input consists of a 3-dimensional image with four channels (one for each modality). Our first layer is a non-trained convolutional layer with 72 $240 \times 240 \times 155 \times 4$ filters, 3-dimensionally convolved over all channels with our input image. In this layer, our filters are the hand-selected Difference of Gaussian filters for a specified modality with all other values in the other three channels being zero. Our training does not go through this layer.

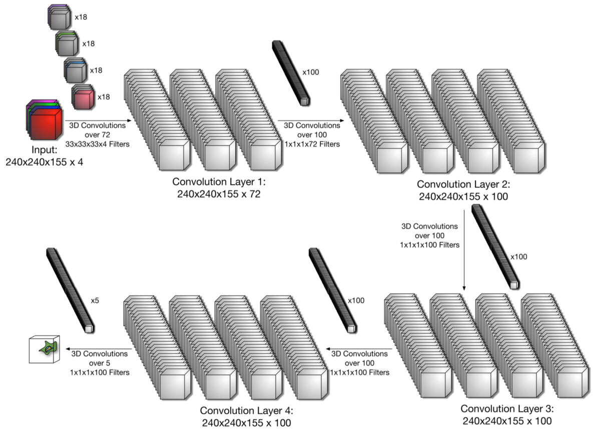


Figure 3: **Main Methodology of Learning Pipeline.** We start with our four modalities of input images. To that, we will convolve our 72 sparse 3-dimensional Difference of Gaussian filters to get our first convolution layer. From that point on, we will be convolving a trivial 1 × 1 × 1 (so a scalar) filter over all channels to get the following convolutional layer. The convolution of a 1 × 1 × 1 filter essentially decouples each pixels information from each other after the initial layers convolution. On the last layer, we will get five channels, each channel giving us a probability score for every pixel in our image for each of the five categories.

The following convolutional layers use convolutions of filters sized 1 × 1 × 1 over the number of channels in the preceding layer (either 72 for the first layer or 100 for all other layers). The 1 × 1 × 1 filter is a scalar that we will constantly multiply over the whole image. In a sense, by only wiring the convolution layers onward with these 1 × 1 × 1 filters, we are decoupling all of the pixels in our image. Because the first convolution layer couples neighborhood information via the 33 × 33 × 33 convolutions, all subsequent layers pixels will maintain some level of information of its original neighborhood. Thus, we are able to decrease the number of weights we would have to train while drastically increasing our effective training data size.

On the last layer, we will get five channels, each channel giving us a probability score for every pixel in our image along the five sub-regions (0 = non-tumor, 1 = necrosis, 2 = edema, 3 = non-enhancing, and 4 = enhancing). We use the softmax function as the loss function. To train the convolutional neural network we use stochastic gradient descent by patient. During test time, we classify using a voting metric that assigns a voxel to one of the five categories based on its classification scores (see algorithm 1). This method is equivalent to how the expert segmentations were pooled from 11 different radiologist's segmentations.

Essentially, our classification priority in descending order goes: enhancing, necrosis, non-enhancing, and edema. If our neural nets returns positive classifications for multiple tumor subtypes, we classify to the positive subtype with the highest priority. This hierarchical design is based off of the hierarchical majority vote used to combine several different algorithmic results [4].

This seemingly arbitrary methodology makes perfect sense in the context of our classification problem. Tumor segmentations are judged generally in terms of three accuracies: whole tumor accuracy, tumor core accuracy, and enhancing tumor accuracy. Thus, because they have their own accuracy scores, we must prioritize classification of the core over the non-core (edema), and then also the enhancing core over the other core. The enhancing core generally covers a smaller area of the brain, which lends even more reason to be more sensitive to its detection.

Results are reported as the standard Dice score calculated via 10-fold cross validation (see: beginning of next section). We do not use cross validation to select parameters, deciding to keep our neural net parameters set to default values. This is both because the additional computation time would be prohibitive, and also because our Dice scores (which are also calculated from the cross validation) would become biased upwards.

Algorithm 1 Hierarchical majority decision Our neural network outputs a number between 0 and 1 per voxel for each of the four tumor structures (edema, non-enhancing, necrosis, enhancing), respectively indicated by p_{edema}, p_{nonenh}, $p_{necrosis}$, p_{enh}

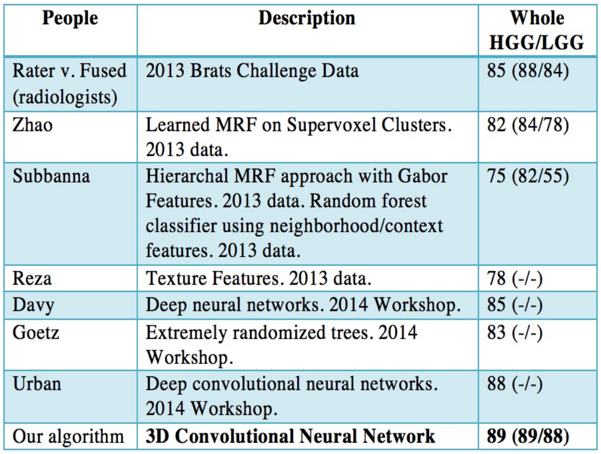
```
procedure HIERARCHICALDECISION($Score$)
    $label \leftarrow NormalTissue$
    if $p_{edema} >= 0.5$ then
        $label \leftarrow Edema$
    end if
    if $p_{nonenh} >= 0.5$ then
        $label \leftarrow Non-Enhancing$
    end if
    if $p_{necrosis} >= 0.5$ then
        $label \leftarrow Necrosis$
    end if
    if $p_{enh} >= 0.5$ then
        $label \leftarrow Enhancing$
    end if
end procedure
```

4. Results

We use Dice Coefficient to compute our results (also known as the Sørensen-Dice Index) [9].

$$DiceScore = \frac{2|Pred \cap Ref|}{|Pred| + |Ref|}$$

where $Pred$ is the set of voxels designated as the region of interest in the prediction and Ref is the set of voxels designated as the region of interest in the reference for comparison (in our case, the expert segmentation).



People	Description	Whole HGG/LGG
Rater v. Fused (radiologists)	2013 Brats Challenge Data	85 (88/84)
Zhao	Learned MRF on Supervoxel Clusters. 2013 data.	82 (84/78)
Subbanna	Hierarchal MRF approach with Gabor Features. 2013 data. Random forest classifier using neighborhood/context features. 2013 data.	75 (82/55)
Reza	Texture Features. 2013 data.	78 (-/-)
Davy	Deep neural networks. 2014 Workshop.	85 (-/-)
Goetz	Extremely randomized trees. 2014 Workshop.	83 (-/-)
Urban	Deep convolutional neural networks. 2014 Workshop.	88 (-/-)
Our algorithm	**3D Convolutional Neural Network**	**89 (89/88)**

Figure 4: **Performance Comparison.** Comparison of our 3D CNN method to the leading methods of both 2013, 2014, radiologists, and our own 2D implementation of our algorithm. The main metric used for our accuracy is Dice Score. Whole = (Edema, Non-enhancing, Necrosis, Enhancing)

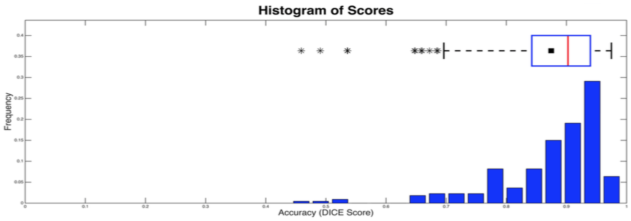


Figure 5: **Histogram of Dice Score Accuracies**

The median Dice performance on the whole tumor detection is 89%! To wit, the inter-radiologist repeatability is only 85%, so our accuracy has saturated with respect to the ground truth. Figure 4 shows a comparison amongst the leading algorithms in the literature. The first row shows the performance of human radiologists. The full distribution of accuracy of the algorithm among the 274 patient is in figure 5. One particularly successful segmentation can be seen in figure 6.

5. Discussion

Our median performance on the whole tumor detection revolves around a dice score of 89, which is right before losing clinical significance. This is very competitive with previous methods. One clear issue with our algorithm is dealing with outliers as can be seen in figure 5. More than half of our segmentations are wildly successful, but some segmentations return sub-50% scores, which you would not typically see with a radiologist.

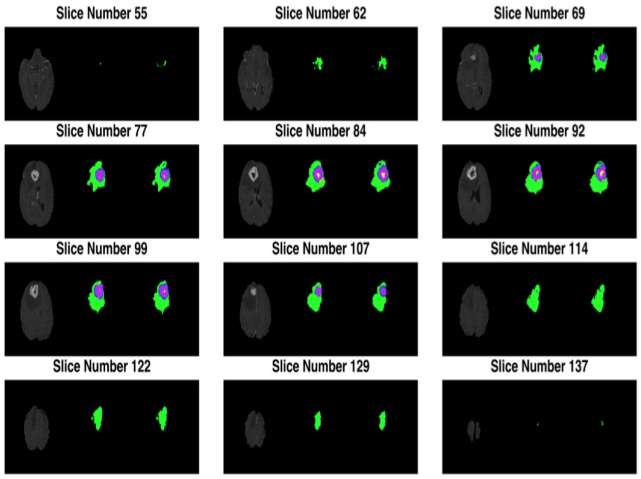


Figure 6: **Slices-by-Slices Result** A representative slice view of a test case giving 98% Dice score accuracy. Each of the 12 images has three parts. On the left is the T1 post-contrast image at the given slice, the middle image is the expert segmentation labeling, and the right-hand side image is the labelling at the slice by our algorithm. The label color scheme follows that which was introduced in figure 1.

figure 4 shows a comparison amongst the top results in the literature. Some notable ones in 2013 include ones by Zhao and Subbanna which incorporated Markov Random Fields (MRF), achieving a Dice accuracy of 82, respectively [4]. Festa from 2013 used random forests to achieve a Dice of 62 [4]. In 2014, groups used deep learning and convolution neural nets (CNNs) to achieve accuracies of 85 (Davy) and 88 (Urban) [10]. Our method improves upon the existing vanilla convolutional neural network architecture by making some high biased assumptions that allow us to drastically reduce our variance.

Basically, one of our assumption is that voxels are independent, only coupled by information tied to their feature vectors. Certainly, this is a high-bias assumption, but it allows us to use $n = 4.5$ billion training examples rather than only $n = 274$ patient samples. Neural nets are ultimately low-bias systems, but our hope is that the improved variance caused by the enlarged sample space will overcompensate for our high-bias assumptions. Furthermore, we contrast our method with standard deep learning algorithms like CNN, which learn the features using convolution kernels on the inputs. Vanilla CNNs use each patient as a single training sample. We, on the other hand, select Difference of Gaussian (DoG) convolution filters to relate voxels to their neighborhoods. Although we are highly biasing our system compared to usual deep learning framework, our features may be more adequate since our subsequent feed-forward neural net can then learn higher level features from each pixel's lower level features.

6. Conclusion

The higher-biased CNN 3-D architecture performs remarkably well on the glioblastoma segmentation problem. The segmentation results are competitive with those using much more complex methods, and we argue our success is due to our smart choice of features along with a greatly enlarged sample space and flexible training method (neural nets). Our algorithm is powerful despite its relatively high bias, and we hope that it may serve the medical community in their work.

The natural next step of this project is a thorough analysis of our models asymptotes. We have claimed that our large data set has significantly reduced model variance, but it is unknown whether we can further reduce variance with more data. Given that our segmentation algorithm is already on par with our reference expert segmentations, we suspect but would like to confirm that our model has already reached its large-data asymptotic performance.

7. Acknowledgments

This project would not have been possible without the help of our research mentor Darvin Yi and the Gevaert Biomedical Informatics Lab. In addition, we would like to thank the CS231N course staff for their ongoing support.

References

[1] World Cancer Report 2014. World Health Organization. 2014. pp. Chapter 5.16.

[2] Bleeker, Fonnet E., Remco J. Molenaar, and Sieger Leenstra. ”Recent Advances in the Molecular Understanding of Glioblastoma.” J Neurooncol Journal of Neuro-Oncology 108.1 (2012): 11-27. Web.

[3] Gallego, O. ”Nonsurgical Treatment of Recurrent Glioblastoma.” Current Oncology Curr. Oncol. 22.4 (2015): 273. Web.

[4] Menze, B. H. The Multimodal Brain Tumor Image Segmentation Bench-mark (BRATS). IEEE Transactions on Medical Imaging 2014 (2013).

[5] Njeh, Ines, Lamia Sallemi, Ismail Ben Ayed, Khalil Chtourou, Stephane Lehericy, Damien Galanaud, and Ahmed Ben Hamida. ”3D Multimodal MRI Brain Glioma Tumor and Edema Segmentation: A Graph Cut Distribution Matching Approach.” Computerized Medical Imaging and Graphics 40 (2015): 108-19. Web.

[6] Smirniotopoulos, James G., Frances M. Murphy, Elizabeth J. Rushing, John H. Rees, and Jason W. Schroeder. ”Patterns of Contrast Enhancement in the Brain and Meninges1.” RadioGraphics 27.2 (2007): 525-51. Web.

[7] Lawrence, S., C.l. Giles, Ah Chung Tsoi, and A.d. Back. ”Face Recognition: A Convolutional Neural-network Approach.” IEEE Trans. Neural Netw. IEEE Transactions on Neural Networks 8.1 (1997): 98-113. Web.

[8] Krizhevsky, A., Sutskever, I., and Hinton, G.E. ”ImageNet Classification with Deep Convolutional Neural Networks.” Adv. Neural Inf. Process. Syst. (2012) 1-9. Web.

[9] Srensen, T. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. Kongelige Danske Videnskabernes Selskab. (1948) 5 (4):1-34.

[10] Challenge Manuscripts.” (2014) MICCAI 2014. Harvard Medical School, Boston, Massachusetts (2014) .