An Ensemble of 2D Convolutional Neural Network for 3D Brain Tumor Segmentation

Kamlesh Pawar[1,2](✉), Zhaolin Chen[1], N. Jon Shah[1,3],
and Gary F. Egan[1,2]

[1] Monash Biomedical Imaging, Monash University, Melbourne, Australia
kamlesh.pawar@monash.edu
[2] School of Psychological Sciences, Monash University, Melbourne, Australia
[3] Institute of Medicine, Research Centre Juelich, Juelich, Germany

Abstract. We propose an ensemble of 2D convolutional neural networks to predict the 3D brain tumor segmentation mask using the multi-contrast brain images. A pretrained Resnet50 and Nasnet-mobile architecture were used as an encoder, which was appended with a decoder network to create an encoder-decoder neural network architecture. The encoder-decoder network was trained end to end using T1, T1 contrast-enhanced, T2 and T2-Flair images to classify each pixel in the 2D input image to either no tumor, necrosis/non-enhancing tumor (NCR/NET), enhancing tumor (ET) or edema (ED). Separate Resent50 and Nasnet-mobile architectures were trained for axial, sagittal and coronal slices. Predictions from 5 inferences including Resnet at all three orientations and Nasnet-mobile at two orientations were averaged to predict the final probabilities and subsequently the tumor mask. The mean dice scores calculated from 166 were 0.8865, 0.7372 and 0.7743 for whole tumor, tumor core and enhancing tumor respectively.

Keywords: Convolutional neural network · Ensemble networks · Residual learning · Brain tumor segmentation

1 Introduction

Automated brain tumor segmentation [1–6] from magnetic resonance images is a challenging task due to variations in the acquisition protocol at different imaging sites. Different imaging parameters including field strength, acceleration factors, resolution, etc. causes variance in the MR images which makes it difficult for automated algorithms to accurately segment the brain tumor regions. Accurate segmentation of brain tumor or gliomas is an important task in grading and monitoring of the disease progression.

Brain tumor segmentation challenge (Brats) is an annual competition which provides manually segmented brain tumor dataset [7–11] to assess the performance of brain tumor segmentation algorithms. Brats challenge started with brain tumor segmentation and have been extended to the task of survival prediction and quantification of uncertainty in segmentation. In recent times, with the availability of large annotated dataset and compute power, most of the best performing algorithms in the challenge are

A. Crimi and S. Bakas (Eds.): BrainLes 2019, LNCS 11992, pp. 359–367, 2020.
https://doi.org/10.1007/978-3-030-46640-4_34

based on deep learning [12]. The best performing algorithms proposed different realizations of the encoder-decoder neural network architectures. Algorithms based on the variations of 3D Unet [13] have been used in the Brats challenge. Since the implementation of 3D Unet requires a large amount of memory, a patch-based approach is often used. The patch-based approach involves training the 3D Unet on a 3D patch of the image often a cube of 64 or 128 depending on the available memory and width/depth of the network. The large memory requirement of the 3D Unet restricts the width and depth of the network. Contrary a 2D Unet [14] requires comparatively less memory than the 3D counterpart at an expense of loss of information from the third spatial dimension. In 2D Unet each image slice is processed independently without considering any information from the rest of the slices within the volume.

In this work, we improve our previous method [15] and propose to use an ensemble of 2D encoder-decoder networks with each network predicting segmentation probabilities for a different orientation (axial, sagittal and coronal). The predicted probabilities from the different orientation are averaged to predict the final probability maps and the segmentation mask for the whole 3D volume. Since probability maps from each individual encoder-decoder are from a different orientation the final averaged probability may contain the 3D information. We hypotheses that false positives from one orientation will be suppressed, when its predicted probability is averaged with the probability map from another orientation.

2 Methods

2.1 Dataset

Manually segmented dataset of brain tumor MR images was provided by the organizers of BRATS challenge. The dataset consisted of two types of brain tumor images namely high-grade tumor (HGG) and low-grade tumor (LGG). Four different contrast T1, T2, T1 contrast-enhanced and T2 Flair image were provided with the manually segmented masks. The mask consisted of three different labels, the necrotic and non-enhancing tumor core (NCR/NET - label 1), the peritumoral edema (ED - label 2), GD-enhancing tumor (ET - label 4) and everything else is classified as label 0. A total of 335 subjects were present consisting of 259 HGG and 76 LGG. The whole dataset was divided into two, one for training and another for validation. Following was the composition of the training and local validation dataset.

- Training dataset: consisted of 288 HGG and 61 LGG subjects.
- Local validation dataset: consisted of 51 HGG and 61 LGG subjects.

Apart from the local validation dataset, another 125 cases were provided to validate the generalization of the model. These 125 cases were provided without the ground truth and segmentation performance was evaluated online using the CBICA Image Processing Portal (https://ipp.cbica.upenn.edu).

2.2 Network Architecture

Our approach consisted of using a 2D convolutional neural network on the individual slices of the whole 3D brain image. We performed end to end training with input being multi-contrast brain images (T1, T2, Tl-CE, T3-Flair) and output being the segmentation mask. The overview of the segmentation process is depicted in Fig. 1, we used an ensemble of 2D networks to predict the segmentation of the whole 3D volume.

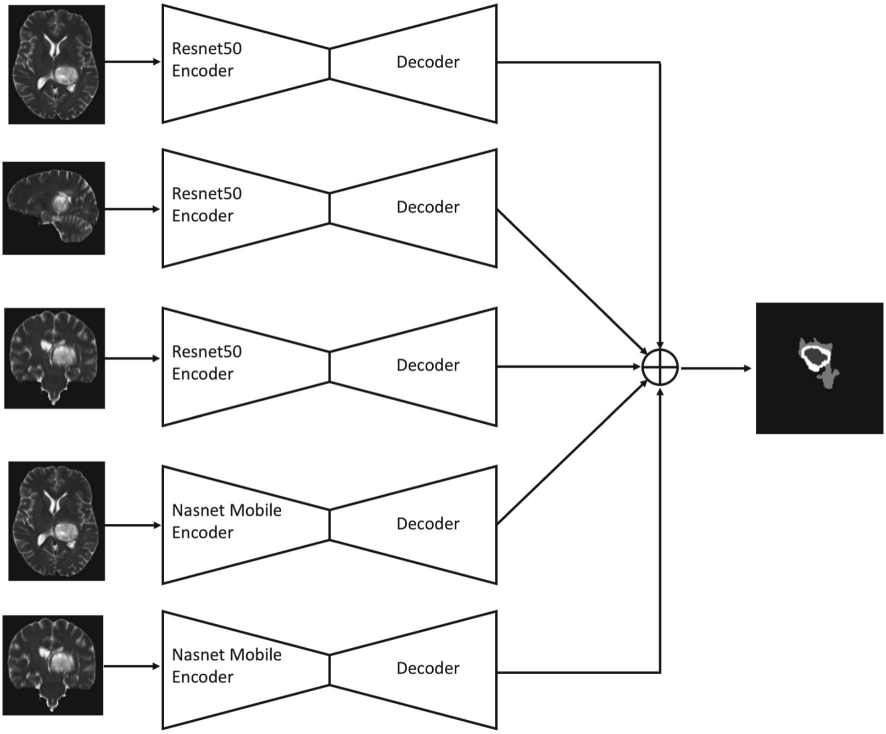

Fig. 1. Overview of the segmentation process. Six separate networks were trained consisting of three Resne50 encoder-decoder architecture for axial, sagittal and coronal orientations and three Nasnet mobile encoder-decoder architecture for axial, sagittal and coronal orientations. The probabilities from the individual predictions were averaged and the segmentation mask was generated from the averaged probabilities.

The encoder-decoder architecture similar to the Unet was the building blocks of the ensemble network. Specifically, we used five separately trained encoder-decoder networks in the ensemble. The encoder-decoder architectures consisted of three networks with Resnet50 [16] as encoder and two networks with Nasnet-mobile [17] as an encoder. A spate encoder-decoder network was trained for each orientation (axial, sagittal and coronal).

The decoder part in each of the network was the same and consisted of a series of convolution and upsampling operations. One block of a decoder is depicted in Fig. 2, which consists of a 2D upsampling operation by a factor of 2 using bilinear interpolation. The upsampling layer increases the spatial dimension of the features using bilinear interpolation and increased size features are then concatenated with the features from the encoder part having the same spatial dimension. The concatenated features are then passed through the two blocks of convolution, batch normalization, spatial dropout and rectilinear activation (ReLU). The number of features for each convolutional layer in the decoder was 256 at each scale except the last scale where it was 128. The convolution kernel size was always 3 × 3 in the decoder network. The last layer of the decoder network consisted of four features, a softmax activation was applied on the last layer, which converts the features into probability maps corresponding to the four classes (NCR/NET, ED, ET or no tumor).

As depicted in Fig. 1, at the time of inference individual slices of the 3D image at different orientations, were processed through the 2D encoder-decoder networks. For Resnet50 predictions were made for axial, sagittal and coronal orientations while for Nasnet-mobile predictions were made for axial and coronal orientations. With Nasnet-mobile encoder-decoder we did not find performance improvement with the sagittal orientation hence it was not used for inference on sagittal orientation. The predicted probabilities from individual 2D slices were stacked to form a 3D volume and 3D volumes from all orientations were averaged for the whole 3D volume. An argmax along the channel dimension on the averaged probability map classified each pixel into one of the four classes (NCR/NET, ED, ET or no tumor).

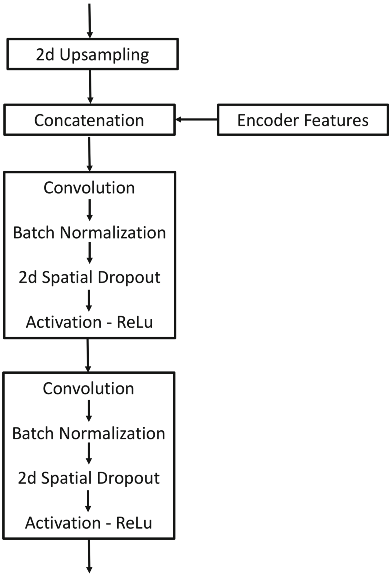

Fig. 2. One block of the decoder network, which first upsamples the input features by a factor of 2 using bilinear interpolation and concatenate the upsampled features with the same scale features form the encoder network. The concatenated features are passed through two blocks of convolution with batch normalization and ReLU activation.

2.3 Pre-processing

The spatial dimension of the input image was 240 × 240 × 155. However, all the 2D networks were trained on 256 × 256 images. The 2D input images were first zero-padded symmetrically to make the 2D input to be 256 × 256.

Since the data was sourced from multiple sites, a preprocessing is required to normalize the images. We used a simple pre-processing of normalizing the mean and standard deviation of the whole 3D volume to zero mean and unity standard deviation using Eq. 1.

$$x_{pp} = \frac{(x - \bar{x})}{std(x)} \tag{1}$$

where x_{pp} is the preprocessed 3D volume, x is the input 3D volume, $\bar{x}$ is the mean of the input volume and $std(x)$ is the standard deviation of input.

2.4 Training

The training of the network was performed on the Keras [18] deep learning library with Tensorflow backend. The adaptive stochastic gradient descent Adam optimizer was used for training the network with a batch size of 4 and initial learning rate of 0.0001. We considered the training of 2000 batches as one epoch. The learning rate was decreased with a step decay of 0.96 per epoch. All the networks were trained for 100 epochs and the network for which the average dice score was maximum on the local validation dataset was chosen as the best model and used for inference on the no ground truth validation dataset.

The loss function used to train the Resnet encoder-decoder architecture consisted of a weighted sum of categorical cross-entropy and soft dice loss. The soft dice loss is defined as:

$$dice\_loss = \frac{2 * \sum p_p * p_t}{\sqrt{\sum p_p^2 * p_t^2}} \tag{2}$$

where p_p is the predicted probability map and p_t is the true probability map.

We trained the Resnet50 encoder-decoder with the weighted sum of categorical cross-entropy loss and dice loss with a weight of 1.0 for cross-entropy and weight of 0.1 for dice loss. For the Nasnet-mobile encoder-decoder, only categorical cross-entropy was used as a loss function.

The results of segmentation were evaluated using the dice score, sensitivity (true positive rate) and specificity (true negative rate) and Hausdorff distance (95%). The evaluation on the validation dataset was calculated using online web-portal provided by the BRATS organizers.

3 Results and Discussion

The individual predictions of the trained networks were ensemble as depicted in Fig. 1 and the segmentation masks were uploaded to the online validation portal. Figure 3 shows the dice scores for the 125 validation subjects and 166 test subjects calculated by the online portal. The median dice scores were higher than the mean dice scores for both the test and validation dataset, suggesting that few difficult cases were segmented by the network with lower accuracy. The dice scores for the test dataset were higher than the validation dataset and also the sample size for the test dataset was larger 166 compared to 125 for the validation dataset. Higher dice score for test dataset (sample size 166) suggests that the algorithm works well for most of the cases but does require further improvements to accurately predict the segmentation for a few subjects that were segmented with lower accuracy.

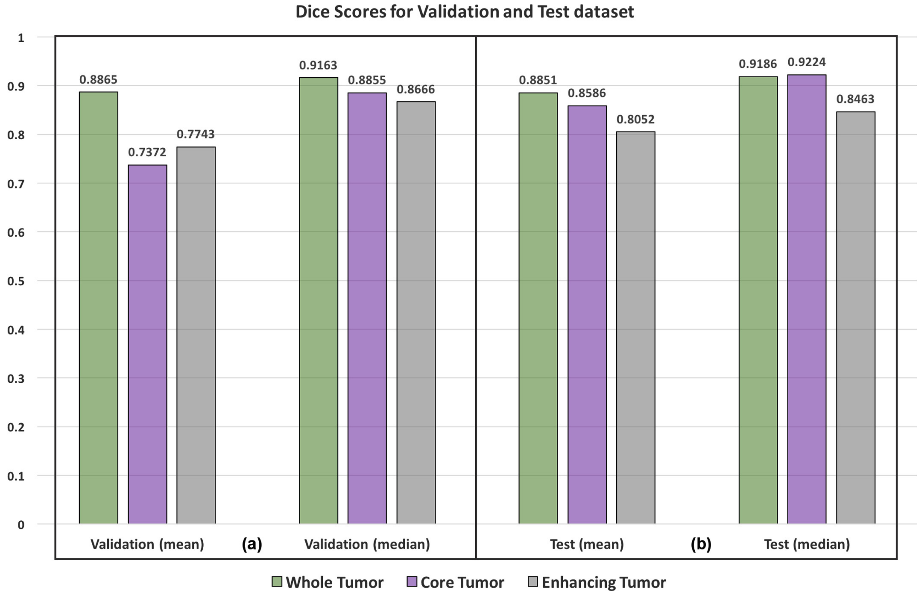

Fig. 3. Bar plot showing the mean dice score using the proposed method; **(a):** bar plot for validation dataset from 125 subjects **(b):** bar plot for test dataset from 166 subjects

Table 1 shows the mean and median dice score, sensitivity and Hausdorff distance on the 125 validation subjects and Table 2 shows the mean and median dice score and Hausdorff distance for 166 test subjects.

Table 1. Quantitative score on validation dataset of 125 subjects calculated using the online IPP portal

	Whole tumor	Core tumor	Enhancing tumor
Dice score (mean)	0.8865	0.7372	0.7743
Dice score (median)	0.9163	0.8855	0.8666
Hausdorff distance	4.2348	5.7720	8.1844
Sensitivity (mean)	0.8602	0.6996	0.7786
Sensitivity (median)	0.9054	0.8510	0.8510

Table 2. Quantitative score on test dataset of 166 subjects calculated using the online IPP portal

	Whole tumor	Core tumor	Enhancing tumor
Dice score (mean)	0.8851	0.8586	0.8052
Dice score (median)	0.9186	0.9224	0.8463
Hausdorff distance	6.4109	4.6700	3.4515

The medians of the dice scores are higher than that of the mean for all three categories of the tumor. The higher median indicates that there are a few harder cases where the algorithm fails to perform well. Usually, the performance on the enhancing tumor class is more challenging compared to the other two classes. However, it is worthwhile to note that the performance of our algorithm on the enhancing tumor class is comparatively higher compared to the tumor core class. This suggests that there is a scope of improvement for the core tumor class, which may require further training and fine-tuning of the network. Two representative segmentations are shown in Fig. 4, one for a highly accurate prediction with average dice score of 0.9508 (Fig. 3 (a)) and another for less accurate segmentation with average dice score of 0.6301 (Fig. 3 (b)).

This work aimed to reduce the memory footprints of the 3D networks by transforming it into multiple 2D networks. This transformation constitutes a trade-off between the computational complexity and memory requirements, the proposed approach reduces the memory footprints but increases the computational complexity. For instance, a 3D network of similar architecture would require 3 times more computation compared to 2D counterpart. However, an ensemble of five 2D networks makes the computational complexity to be 5 times than the single 2D network.

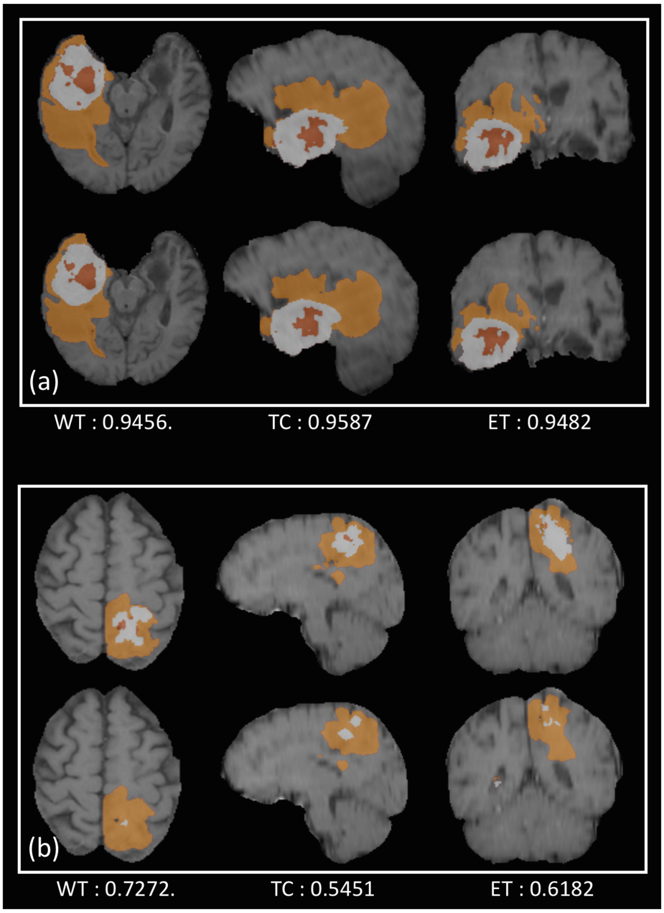

Fig. 4. Segmentation results for two representative images with red color: NCR/NET, orange color: edema and white color: ET. The bottom of the figure shows the dice score for whole tumor (WT), tumor core (TC) and enhancing tumor (ET) respectively. **(a):** first row shows ground truth segmentation; second row shows predicted segmentation results for one of the highly accurate prediction; **(b):** first row shows ground truth segmentation; second row shows predicted segmentation results for one of the less accurate prediction. (Color figure online)

4 Conclusion

In this work, we have presented an approach to predict the brain tumor segmentation for the whole 3D volume using an ensemble to the 2D CNN. Specifically, we used resnet50 and nasnet-mobile architectures for the predictions. The results are promising with the average dice score of 0.8851, 0.8586 and 0.8052 for whole tumor, core tumor and enhancing tumor respectively.

References

1. Sharma, N., Aggarwal, L.M.: Automated medical image segmentation techniques. J. Med. Phys./Assoc. Med. Phys. India **35**, 3 (2010)
2. Pham, D.L., Xu, C., Prince, J.L.: Current methods in medical image segmentation. Annu. Rev. Biomed. Eng. **2**, 315–337 (2000)
3. Corso, J.J., Sharon, E., Dube, S., El-Saden, S., Sinha, U., Yuille, A.: Efficient multilevel brain tumor segmentation with integrated bayesian model classification. IEEE Trans. Med. Imaging **27**, 629–640 (2008)
4. Angelini, E.D., Clatz, O., Mandonnet, E., Konukoglu, E., Capelle, L., Duffau, H.: Glioma dynamics and computational models: a review of segmentation, registration, and in silico growth algorithms and their clinical applications. Curr. Med. Imaging Rev. **3**, 262–276 (2007)
5. Gupta, M.P., Shringirishi, M.M.: Implementation of brain tumor segmentation in brain mr images using k-means clustering and fuzzy c-means algorithm. Int. J. Comput. Technol. **5**, 54–59 (2013)
6. Liu, J., Li, M., Wang, J., Wu, F., Liu, T., Pan, Y.: A survey of MRI-based brain tumor segmentation methods. Tsinghua Sci. Technol. **19**, 578–595 (2014)
7. Bakas, S., et al.: Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features. Sci. Data **4**, 170117 (2017)
8. Bakas, S., et al.: Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge. arXiv preprint arXiv:1811.02629 (2018)
9. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-LGG collection. The Cancer Imaging Archive 286 (2017)
10. Menze, B.H., et al.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE Trans. Med. Imaging **34**, 1993–2024 (2015)
11. Bakas, S., et al.: Segmentation labels and radiomic features for the pre-operative scans of the TCGA-GBM collection. The Cancer Imaging Archive (2017)
12. LeCun, Y.A., Bengio, Y., Hinton, G.E.: Deep learning. Nature **521**, 436–444 (2015)
13. Kamnitsas, K., Ledig, C., Newcombe, V.F., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D., Glocker, B.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. Med. Image Anal. **36**, 61–78 (2017)
14. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, William M., Frangi, Alejandro F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
15. Pawar, K., Chen, Z., Shah, N.J., Egan, G.: Residual encoder and convolutional decoder neural network for glioma segmentation. In: Crimi, A., Bakas, S., Kuijf, H., Menze, B., Reyes, M. (eds.) BrainLes 2017. LNCS, vol. 10670, pp. 263–273. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75238-9_23
16. He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the CVPR IEEE, pp. 770–778 (2016)
17. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.V.: Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 8697–8710 (Year)
18. Chollet, F.: Keras (2015)