ACTA UNIV. SAPIENTIAE, INFORMATICA 10, 1 (2018) 110–132

DOI: 10.2478/ausi-2018-0007

Low and high grade glioma segmentation in multispectral brain MRI data

László SZILÁGYI
Sapientia University
Târgu Mureş, Romania
email: lalo@ms.sapientia.ro

David ICLĂNZAN
Sapientia University
Târgu Mureş, Romania
email: iclanzan@ms.sapientia.ro

Zoltán KAPÁS
Sapientia University
Târgu Mureş, Romania
email: zoltankapas@yahoo.com

Zsófia SZABÓ
Sapientia University
Târgu Mureş, Romania
email: sz_zsokaa@yahoo.com

Ágnes GYŐRFI
Sapientia University
Târgu Mureş, Romania
email: gyorfiagnes@ms.sapientia.ro

László LEFKOVITS
Sapientia University
Târgu Mureş, Romania
email: lefkolaci@ms.sapientia.ro

Abstract. Several hundreds of thousand humans are diagnosed with brain cancer every year, and the majority dies within the next two years. The chances of survival could be easiest improved by early diagnosis. This is why there is a strong need for reliable algorithms that can detect the presence of gliomas in their early stage. While an automatic tumor detection algorithm can support a mass screening system, the precise segmentation of the tumor can assist medical staff at therapy planning and patient monitoring. This paper presents a random forest based procedure trained to segment gliomas in multispectral volumetric MRI records. Beside the four observed features, the proposed solution uses 100 further

Computing Classification System 1998: I.5.3
Mathematics Subject Classification 2010: 68T10
Key words and phrases: machine learning, random forest, image segmentation

features extracted via morphological operations and Gabor wavelet filtering. A neighborhood-based post-processing was designed to regularize and improve the output of the classifier. The proposed algorithm was trained and tested separately with the 54 low-grade and 220 high-grade tumor volumes of the MICCAI BRATS 2016 training database. For both data sets, the achieved accuracy is characterized by an overall mean Dice score $> 83\%$, sensitivity $> 85\%$, and specificity $> 98\%$. The proposed method is likely to detect all gliomas larger than $10\,\mathrm{mL}$.

1 Introduction

Most brain tumors are diagnosed in a certain advanced stage, when the symptoms convince the patient to go to the doctor. The average brain tumor patient lives 14 months after the diagnosis. The development of imaging devices and computers make it possible to elaborate intelligent automated procedures that would allow for regular screening of a larger population and detecting most tumors in an earlier phase. Beside establishing the diagnosis, the automatic segmentation and quantitative analysis can assist therapy planning and evolution tracking of the tumor. However, automatic tumor segmentation is not only utmost important task, but also a very challenging one, because of the high variety of anatomical structures and low contrast of current imaging techniques, which make the difference between normal regions and the tumor hardly recognizable for the human eye [6].

Magnetic resonance imaging (MRI) is the preferred imaging device in brain tumor screening, due to its better contrast and relatively fine resolution. However, it also bears difficulties like the possible presence of intensity inhomogeneity [27], and the relative intensity values that vary from device to device and from patient to patient [19]. The MICCAI Brain Tumor Segmentation Challenge, organized yearly since 2012, intensified the research in this topic and led to several important solutions, which are usually assisted by the use of prior information, and employ various image processing and pattern recognition methodologies. Asman et al. [2] applied a non-parametric intensity analysis in combination with a segmentation based on multiple atlases. Ghanavati et al. [5] provided a solution using the AdaBoost classifier to distinguish tumor voxels from normal ones using features based on intensity, texture, and symmetry. Hamamci et al. [7] proposed a cellular automata driven method that produces segmentation based on level sets. Sachdeva et al. [22] deployed a content based active contour model relying on intensity and texture features extracted from the histogram and co-occurrence matrix of the MRI data. Njeh et al. [18] introduced a graph cut based solution that performs distribution matching, which is highly efficient because of using rather global than pixelwise information. Zhang et al. [29] proposed a support vector machine based procedure to follow the evolution of brain tumors over time. Tustison et al. [26] combined random forests with symmetry based features to segment brain tumors. Szilágyi et al. [25] provided a semi-supervised framework for the fuzzy c-means clustering algorithm to produce accurately segmented tumors. Kanas et al. [13] combined a clustering based preprocessing with a multi-parametric random walker segmentation. Havaei et al. [8] developed an automatic brain tumor segmentation procedure based on deep neural networks that exploits both local and global contextual features simultaneously. Pereira et al. [20] proposed a convolutional neural network solution exploiting small kernels and successfully applied it for brain tumor segmentation. Menze et al. [17] combined a Gaussian mixture model with the expectation maximization (EM) algorithm to achieve an accurate segmentation. Another Gaussian mixture based accurate solution was given by Juan-Albarracín et al. [12]. Islam et al. [11] employed multifractional Brownian motion features to provide patient-independent characterization of tumor tissues and applied the AdaBoost algorithm for tissue segmentation. Shin et al. [23] proposed deep convolutional neural networks and successfully combined it with transfer learning. Huang et al. [9] provided a brain tumor segmentation framework employing local independent projection-based classification. Lê et al. [15] proposed a brain tumor segmentation procedure based on a tumor growth model. Pinto et al. [21] employed extremely random trees to provide a hierarchical solution to the low-grade glioma segmentation problem. Zaouche et al. [28] provided a semi-supervised low-grade glioma segmentation based on specially designed spatial edge filters and maximum likelihood optimization. For further information on current brain tumor segmentation techniques, there are available recent reviews [6, 10].

In a previous paper [14] we have presented a preliminary study on the use of random forests in the detection and segmentation of high-grade gliomas. Each voxel was characterized by a 16-element feature vector, including minimum, maximum, and median values computed from the neighborhood of the voxel. The procedure proposed in that study was evaluated using the 220 high-grade tumor volumes from the MICCAI BRATS 2016 data set. The best overall Dice Score was found 81%. As a further development [24], we proposed a random forest solution trained and tested using 104-element feature vectors that included various computed morphological and Gabor wavelet features. Our main goal in this paper is to perform a detailed evaluation of the solution based on 104-element feature vector using all low-grade (LG) and high-grade

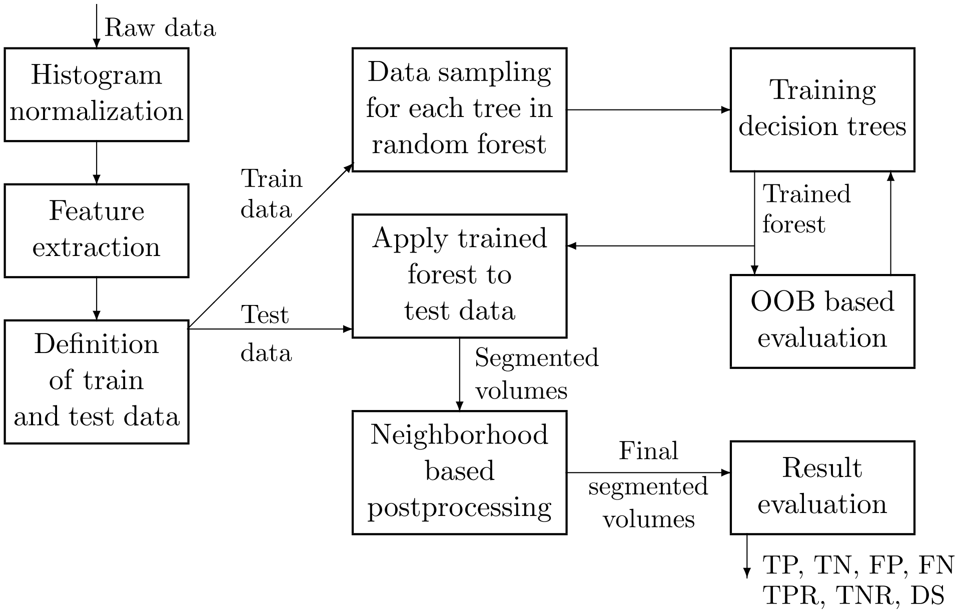

Figure 1: Block diagram of the proposed method.

(HG) tumor volumes of the MICCAI BRATS 2016 train database.

The rest of this paper is structured as follows: Section 2 gives details on the proposed methodology. Section 3 exhibits and discusses the achieved results. Finally, Section 4 concludes the investigation.

2 Materials and Methods

Our goal was to elaborate an accurate segmentation procedure based on a machine learning algorithm, applicable for both LG and HG brain tumor volumes separately. This paper presents results obtained using a random forest approach, combined with histogram normalization, Gabor feature extraction for texture characterization, and a neighborhood-based post-processing. The trees of the random forest are trained to separate the whole tumor from normal tissues. The structure of the elaborated segmentation procedure is presented in Fig. 1.

Property	LG volumes	HG volumes
Number of data volumes	54	220
Average size of whole tumor (mL)	101.1	110.6
Minimum size of whole tumor (mL)	18.9	8.5
Maximum size of whole tumor (mL)	265.3	318.4
Total number of negative voxels	72.2M	311.5M
Total number of positive voxels	5.46M	24.32M

Table 1: Main attributes of the input data set

2.1 MICCAI BRATS data sets

Fully anonymized multimodal MR image data was obtained from the MICCAI 2016 Challenge on Multimodal Brain Tumor Segmentation [16]. This database contains multi-contrast MR scans of 274 glioma patients, out of which 54 having low-grade and 220 having having-grade glioma lesions. For each patient, multimodal ($T1$, $T2$, FLAIR, and post-Gadolinium $T1$ ($T1C$)) MR images were recorded and linearly co-registered to the T1 contrast image. All data volumes were skull stripped, and interpolated to $1\ mm$ isotropic resolution. Each record contains approximately 1.5 millions of true tissue voxels, out of which the rate of positives ranges from 0.5% to 20%. Manual annotation produced by human experts is provided for all voxels. Further technical details of the data set are given in Table 1.

2.2 Data preprocessing

A major drawback of MR imaging consists in the lack of a standard scale of image intensities. This is why we need to map the histogram of each data channel of BRATS volumes onto a uniform scale. Although literature contains various recommendation is this issue [19, 4], we opted to employ a simple linear transform $x \rightarrow \alpha x + \beta$ to all intensities, where parameters α and β were established separately for each volume and each data channel such a way that the 25-percentile and 75-percentile values became 600 and 800, respectively. Further on, a minimum and a maximum intensity barrier was enforced at 200 and 1200, respectively. This approximately corresponds to a 10-bit resolution in each data channel.

Most voxels of the MRI records have valid nonzero intensity in all data channels. However, there are voxels with one or more missing values. We considered

Neighborhood	$3 \times 3 \times 3$	3×3	5×5	7×7	9×9	11×11	Total
Average	4	4	4	4	4	4	24
Maximum	4						4
Minimum	4						4
Median		4	4	4	4	4	20
Gradient				16			16
Gabor wavelet						32	32
Total	12	8	8	24	8	40	100

Table 2: Inventory of computed features. All four data channels were involved equally.

that the region of interest (ROI) in the BRATS volumes includes all voxels that have at least one nonzero value in any of the observed data channels. Missing values were replaced by the mean intensity value of existing neighbors within the 26-element immediate spatial neighborhood, or the grand mean of the given data channel whenever no neighbors with correct intensity were found in the neighborhood.

Although the four observed features of each voxel bear a lot more information than any one of them, there is an acute need to extend the feature vectors with further computed features. A total number of 100 computed features were added to the feature vector describing each voxel, according to the inventory given in Table 4. For each of the four observed intensities (T1, T2, T1C, FLAIR), six average, five median, one minimum, one maximum, four gradient values, and further eight Gabor features were extracted. All computed feature values were linearly scaled into the $[200, 1200]$ interval. This way, together with the four observed features, each voxel is described by a 104-element feature vector. These feature vectors are used by the classification stage of the proposed segmentation procedure.

2.3 Data classification

Binary decision trees (BDT) of unlimited depth can describe any hierarchy of crisp (non-fuzzy) two-way decisions [1]. Given an input data set of vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$, where $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,m}]^T$, a BDT can be employed to learn the classification that corresponds to any set of labels $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_n\}$. The classification learned by the BDT can be perfect if there are no identical training vectors with different labels, that is, $\mathbf{x}_i = \mathbf{x}_j$

implies $\lambda_i = \lambda_j$, $\forall i, j \in \{1, 2, \ldots, n\}$. The BDT is built during the training process. Initially the tree consists of a single node, the root, which has to make a decision regarding all n train data vectors. If not all n vectors have the same label, which is likely to be so, then the set of data is not homogeneous, and there is a need for a separation. The decision will compare a single chosen feature, the one with index k ($1 \le k \le m$), of the input vectors with a certain threshold α, and the comparison will separate the vectors into two subgroups: those with $x_{i,k} < \alpha$ ($i = 1 \ldots n$), and those with $x_{i,k} \ge \alpha$ ($i = 1 \ldots n$). The root will then have two child nodes, each corresponding to one of the possible outcomes of the above decision. The left child will further classify those n_1 input vectors, which satisfied the former condition, while the right child those n_2 ones that satisfied the latter condition. Obviously, we have $n_1 + n_2 = n$ with $n_1 > 0$ and $n_2 > 0$. For both child nodes, the procedure is the same as it was for the root. When at a certain point of the learning algorithm, all vectors being classified by a node have the same label λ_p, then the node is declared a leaf node, which is attributed to the class with index p. Another case when a node is declared leaf node is when all vectors to be separated by the node are identical, so there is no possible condition to separate the vectors. In this case, the label of the node is decided by the majority of labels, or if there is no majority, a label should be chosen from the present ones. In our application, this kind of rare leaves are labeled as tumor.

The separation of a finite set of data vectors always terminates in a finite number of steps. The maximum depth of the tree highly depends on the way of establishing the separation condition in each node. Our application uses an entropy based criterion to choose the separation condition. Whenever a node has to establish its separation criterion for a subset of vectors $\overline{\mathbf{X}} \subseteq \mathbf{X}$ containing $\overline{n}$ items with $1 < \overline{n} \le n$, the following algorithm is performed:

1. Find all those features which have at least two different values in $\overline{\mathbf{X}}$.
2. Find all different values for each feature and sort them in increasing order.
3. Set a threshold candidate at the middle of the distance between each consecutive pair of values for each feature.
4. Choose that feature and that threshold, for which the entropy-based criterion
$$E = \overline{n}_1 \log \frac{\overline{n}_1}{\overline{n}} + \overline{n}_2 \log \frac{\overline{n}_2}{\overline{n}} \qquad (1)$$
gives the minimum value, where $\overline{n}_1$ ($\overline{n}_2$) will be the cardinality of the subset of vectors $\overline{\mathbf{X}}_1$ ($\overline{\mathbf{X}}_2$), for which the value of the tested feature is less than (greater or equal than) the tested threshold value.

After having the BDT trained, it can be applied for the classification of test data vectors. Any test vector is first fed to the root node, which according to the stored condition and the feature values of the vector, decides towards which child node to forward the vector. This strategy is followed then by the chosen child node, and the vector will be forwarded to a further child. The classification of a vector terminates at the moment when it is forwarded to a leaf node of the tree. The test vector will be attributed to the class indicated by the labeling of the reached leaf node.

Binary decision trees were trained to separate tumor voxels from negative ones. Due to practical reasons, negative voxels were randomly subsampled to 12% for the BDT training process. Random forests were trained according to the following parameters:

1. The number of trees in the forest denoted by n_T. This parameter was usually set to 255. Experiments proved this number of trees more than necessary for good accuracy.
2. The number of data vectors used to train each tree of the forest, denoted by n_P. Typical values of this parameter ranged from 10 thousand to 500 thousand.
3. The rate (percentage) of negative labeled data within the training set, denoted by p_n.
4. The threshold of positive votes θ_p (expressed in percentage) necessary to assign a voxel to the class of positives. Making a decision according to majority voting would mean using a $\theta_p = 50\%$ threshold, but slightly shifted values of θ may lead to better accuracy.

Ideal parameter settings were identified using the so-called out-of-bag (OOB) data, as recommended by Breiman in [3]. Testing on OOB data allowed us to preselect those forests that were likely to produce high accuracy, and discard those that were prone to severe misclassifications. The best performing trees achieved $93-95\%$ correct decisions, while the most accurately classifying forests scored $96-98\%$ in labeling the OOB data.

2.4 Post-processing

A posterior relabeling scheme was implemented as follows. The input data of the post-processing step consisted in the labels provided by the random forest to all voxels in the test volume. For each voxel, the number of tumor labeled neighbors (ν_T) and the number of all neighbors (ν_{All}) were extracted, using a predefined neighborhood. The final label of a voxel was set to tumor if and only if $\nu_T/\nu_{\text{All}} > \theta_n$. The optimal value of this threshold was established based on tests performed on OOB data. The ideal neighborhood to be employed in post-processing was identified as the cubic $11 \times 11 \times 11$ sized one for LG tumor volumes, and $9 \times 9 \times 9$ for HG tumor volumes.

2.5 Evaluation of accuracy

We employed the Dice score (DS) as the main indicator of accuracy, defined as

$$\mathsf{DS} = \frac{2 \times \mathsf{TP}}{2 \times \mathsf{TP} + \mathsf{FP} + \mathsf{FN}} \in [0, 1] \ , \tag{2}$$

where TP, FP, and FN stand for the number of true positives, false positives, and false negatives, respectively. Fine accuracy is reflected by DS values close to 1, but in this brain tumor segmentation problem, DS values around 0.94 are considered ideal [16], due to inter-rater differences that are present in the ground truth. Further on, sensitivity (or true positive rate, TPR) defined as

$$\mathsf{TPR} = \frac{\mathsf{TP}}{\mathsf{TP} + \mathsf{FN}} \ , \tag{3}$$

specificity (or true negative rate) defined as

$$\mathsf{TNR} = \frac{\mathsf{TN}}{\mathsf{TN} + \mathsf{FP}} \ , \tag{4}$$

and the rate of correct decisions

$$\mathsf{ACC} = \frac{\mathsf{TP} + \mathsf{TN}}{\mathsf{TP} + \mathsf{TN} + \mathsf{FP} + \mathsf{FN}} \tag{5}$$

were used as secondary accuracy indicators, where TN represents the number of true negatives.

If we denote by TP_i, TN_i, FP_i, and FN_i, the true/false positives/negatives obtained at testing volume number i ($i = 1 \dots p$, where p is the number of volumes), then we define average Dice score as

$$\widetilde{\mathsf{DS}} = \frac{1}{p} \sum_{i=1}^{p} \mathsf{DS}_i = \frac{1}{p} \sum_{i=1}^{p} \frac{2 \times \mathsf{TP}_i}{2 \times \mathsf{TP}_i + \mathsf{FP}_i + \mathsf{FN}_i} \ , \tag{6}$$

and overall Dice score as:

$$\overline{\mathsf{DS}} = \frac{2 \times \sum\limits_{i=1}^{p} \mathsf{TP}_i}{2 \times \sum\limits_{i=1}^{p} \mathsf{TP}_i + \sum\limits_{i=1}^{p} \mathsf{FP}_i + \sum\limits_{i=1}^{p} \mathsf{FN}_i} \ . \tag{7}$$

Similarly, we will compute overall and average values for the sensitivity ($\overline{\mathsf{TPR}}$, $\widetilde{\mathsf{TPR}}$) and the specificity ($\overline{\mathsf{TNR}}$,$\widetilde{\mathsf{TNR}}$).

3 Results and Discussion

The proposed algorithm was validated using the total number of 220 HG and 54 LG tumor volumes separately. Both volume sets were ordered randomly and then divided into two groups which were called the even and the odd group. Volumes from the even (odd) group were segmented with random forests trained with all volumes from the odd (even) group. The main parameters of the algorithm were each given several values in a suitable range: train data size between 1k and 500k pixels per tree, rate of negative train data between 85% and 95%. The number of trees was set to 255, but smaller forests were also evaluated via omitting some of the trees. The rate of positive votes (given by the trees of the forest) that is necessary to declare a pixel positive, was also investigated in the range between 20% and 80%, to optimize the final decision. Accuracy results were obtained for all image volumes. Average, median, and overall values of main statistical accuracy indicators were computed as presented in Section 2.5. Table 3 summarizes the main accuracy indicator values for both LG and HG tumor volumes.

Figure 2 presents the overall Dice score values ($\overline{\mathrm{DS}}$) obtained for the output of the random forest, plotted against the training data size, for various values of the negative data rate (p_n) situated between 86% and 95%. In case of LG data, the RF classifier performed best at $p_n = 93\%$, achieving $\overline{\mathrm{DS}} > 81\%$. In case of HG data, the accuracy seems to saturate at $p_n = 94\%$, but the obtained Dice scores are lower. Larger training data size usually leads to better accuracy, but this tendency saturates around 300k pixels per tree in LG data. Figure 3 shows the overall Dice scores obtained after post-processing. Validating the labels given by the random forest to each pixel visibly helped more in HG volumes.

Figures 4 and 5 exhibit the same Dice scores as Figs. 2 and 3, but here the overall Dice scores are plotted against the rate of negatives in the training data p_N, and each curve stands for a certain training data size. These graphs also show us that the random forest classified LG data better than HG data, but the post-processing improved the overall accuracy in HG, thus the final overall Dice scores are close to each other. Post-processing improves the DS by approximately 2% in LG volumes and by 3% in HG volumes. The accuracy difference between the largest and smallest training data size is lower in HG

Accuracy	LG volumes		HG volumes	
Indicator	Before PP	After PP	Before PP	After PP
Overall $\overline{\mathrm{DS}}$	81.0%	83.8%	81.1%	83.6%
Average $\widetilde{\mathrm{DS}}$	77.0%	81.3%	76.2%	80.2%
Median DS	81.0%	84.6%	80.3%	85.5%
Overall $\overline{\mathrm{TPR}}$	76.7%	84.8%	74.5%	83.2%
Average $\widetilde{\mathrm{TPR}}$	77.0%	81.3%	70.7%	77.6%
Median TPR	81.0%	84.6%	75.4%	85.1%
Overall $\overline{\mathrm{TNR}}$	99.03%	98.64%	99.28%	98.86%
Average $\widetilde{\mathrm{TNR}}$	99.04%	98.64%	99.28%	98.86%
Median TNP	99.33%	98.84%	99.55%	99.25%
DS > 80%	30 of 54	42 of 54	114 of 220	144 of 220
DS > 85%	16 of 54	26 of 54	89 of 220	111 of 220
DS > 90%	6 of 54	12 of 54	48 of 220	56 of 220

PP stands for post-processing.

Table 3: Main accuracy indicators

data, probably due to the largest number in data volumes. The optimal rate of negatives in the training data for HG volumes is $p_N = 93\%$ at any tested train data size, while for the LG volumes it varies between 92% and 94%. Below 10k training pixels per tree, there is a sudden drop in accuracy.

Figure 6 exhibits the final overall sensitivity values, for LG and HG volumes separately, plotted against the rate of negatives in the training data (p_N), for training data sizes ranging from 10k to 500k. Sensitivity values have a slight dropping tendency as p_N rises. Sensitivity is higher in HG volumes, especially at smaller training data sizes.

Figure 7 presents the sensibility, specificity, and Dice score values obtained for individual LG and HG records, sorted in increasing order of the accuracy indictors. Sensitivity and Dice score values range between 30% and 100%, but they do not follow a uniform distribution. In both LG and HG data, there is a small subset (10—15%) of volumes which lead to mediocre resuts. Most of these volumes have a lot of missing data. The graphs presented in Fig. 7 suggest that accuracy indicators have a distribution that grants higher median value than the average. Numeric values listed in Table 3 confirm this suggestion. Specificity is over 95% in most of the cases, having its average and median value between 98.5% and 99%. It is important to have such a high specificity,

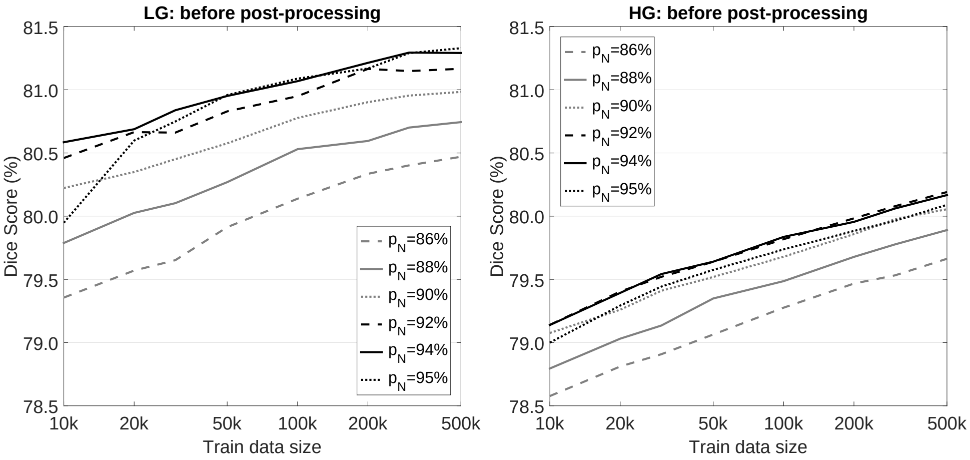

Figure 2: Overall Dice score plotted against the train data size, at various values of the rate of negative train data p_n, and $n_T = 255$ trees in each random forest, without post-processing. Left and right panel exhibit the results obtained for the 54 LG and 220 HG tumor volumes, respectively.

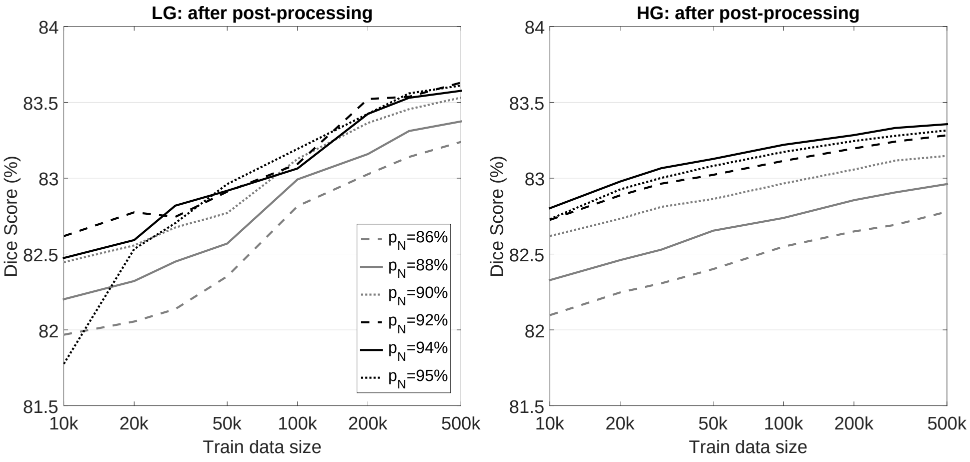

Figure 3: Improved values of the overall Dice scores exhibited in Fig. 2, obtained after post-processing.

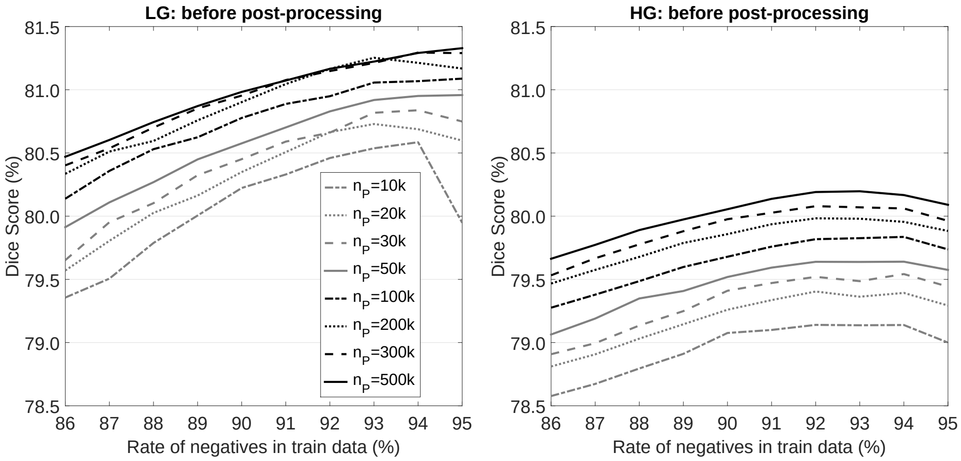

Figure 4: Overall Dice score plotted against the rate of negative train data p_n, at various values of the train data size, and $n_T = 255$ trees in each random forest, without post-processing. Left and right panel exhibit the results obtained for the 54 LG and 220 HG tumor volumes, respectively.

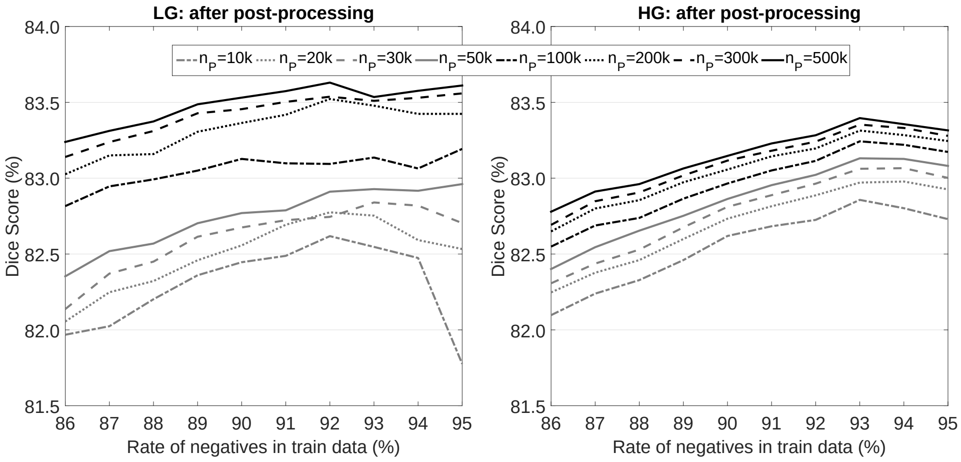

Figure 5: Improved values of the overall Dice scores exhibited in Fig. 4, obtained after post-processing.

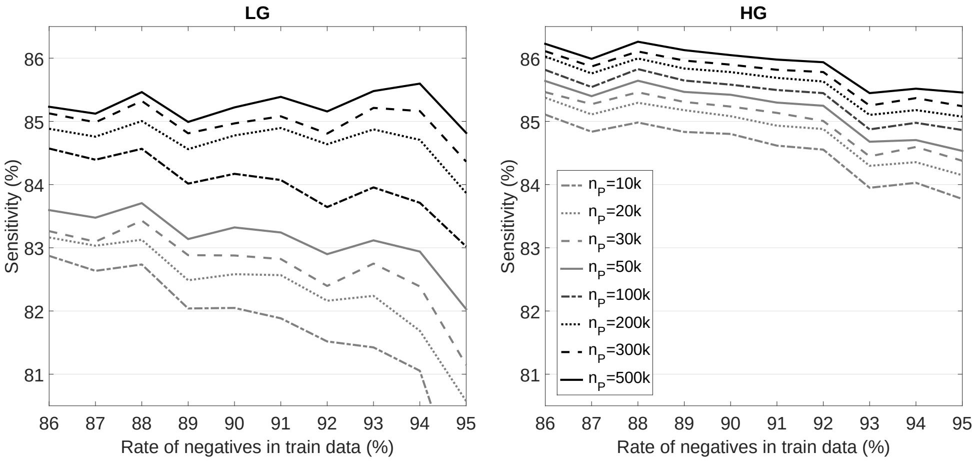

Figure 6: Overall Sensitivity plotted against the rate of negative train data p_n, at various values of the number of pixels used to train each random tree. These results were obtained using $n_T = 255$ trees in each random forest, after post-processing. Left and right panel exhibit the results obtained for the 54 LG and 220 HG tumor volumes, respectively.

because otherwise we would have a lot of false positives due to the high rate of negative data.

Figure 8 depicts the Dice score values obtained for individual MRI records, before and after post-processing, sorted in the increasing order of the Dice score. Left and right panels show LG ang HG volumes, respectively. Post-processing seems to help most volumes, especially those with lower Dice score value. When the random forest classifier produces a Dice score over 90%, post-processing may slightly reduce the accuracy. Figure 9 exhibits similar curves for the Sensitivity instead of the Dice score. Here the post-processing proves highly effective, as all Sensitivity values rise during the last processing step. Dice scores of individual volumes improve by 4% in average for both LG and HG volumes, while Sensitivity values rise by 4% and 7% for LG and HG volumes, respectively. Figure 10 plots individual Dice score values after post-processing vs. before post-processing, for LG volumes in the left panel and HG volumes in the right panel. Figure 11 uses the same representation for Sensitivity values instead of Dice scores. Each cross (×) situated above the diagonal drawn in dashed line indicates a case where post-processing improved

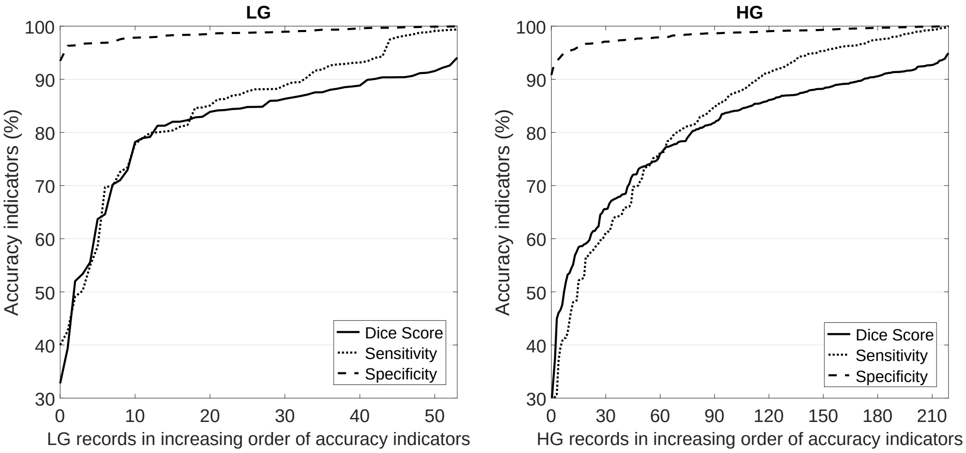

Figure 7: The three main accuracy indicators obtained for each LG (left panel) and HG (right panel) tumor volume separately, after post-processing. Accuracy indicator values were sorted in increasing order.

the value of the accuracy indicator. Figure 10 also contains crosses below the diagonal, which means that the Dice score does not always improve during post-processing. Even if the accuracy is damaged in some cases, the average effect of post-processing is beneficial (Table 3).

Figure 12 plots the individual Dice scores obtained for each LG and HG volume against the size of the tumor, showing the difference between the output of the random forest classifier (left column) and the final post-processed result (right column). There is a general rule that the larger the tumor the better the chances of detection and accurate segmentation. The identified linear trends show that the strongest effect of post-processing occurs in case of small tumors.

Figure 13 shows detailed final result obtained on two MRI slices originating from LG volumes. The top row shows a slice whose segmentation was successful, with a very reduced number of misclassified pixels. The bottom row exhibits a slice where the segmentation of the tumor was visibly mistaken, with a large area of false negatives (unidentified part of the tumor), but the presence of the tumor can be reliably detected based on this segmentation, because most of the tumor pixels were identified.

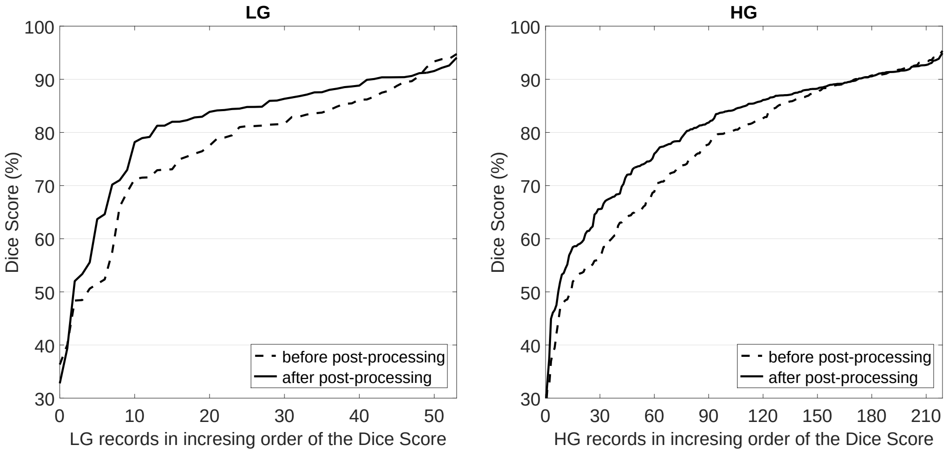

Figure 8: The effect of post-processing upon individual Dice score values obtained for LG (left panel) and HG (right panel) tumor volumes separately. Accuracy indicator values were sorted in increasing order.

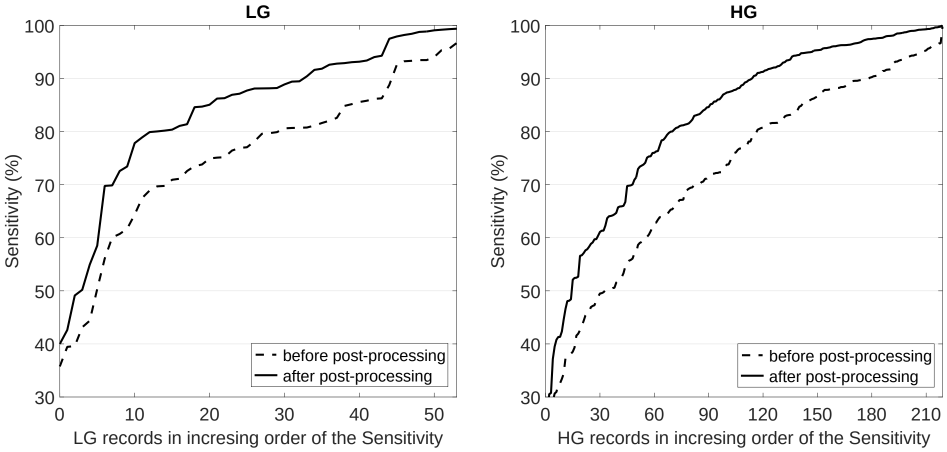

Figure 9: The effect of post-processing upon individual Sensitivity values obtained for LG (left panel) and HG (right panel) tumor volumes separately. Accuracy indicator values were sorted in increasing order.

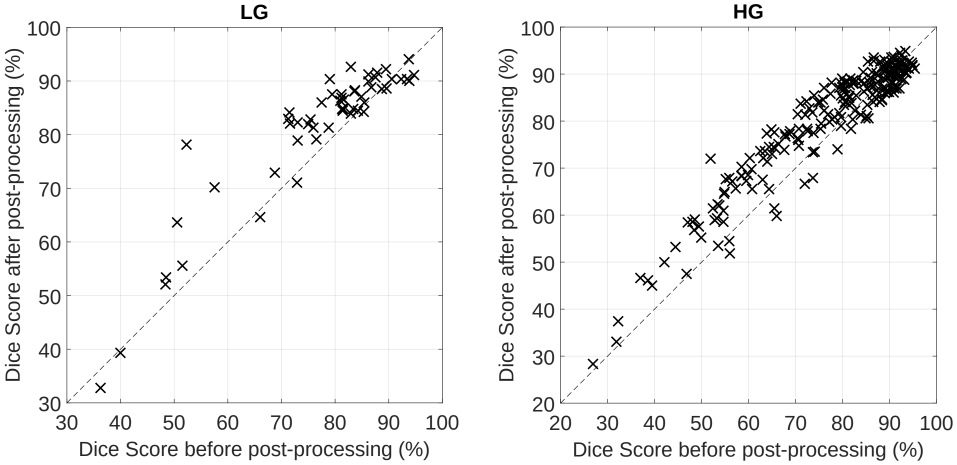

Figure 10: Dice scores obtained for individual LG (left panel) and HG (right panel) tumor volumes, plotted after post-processing vs. before post-processing.

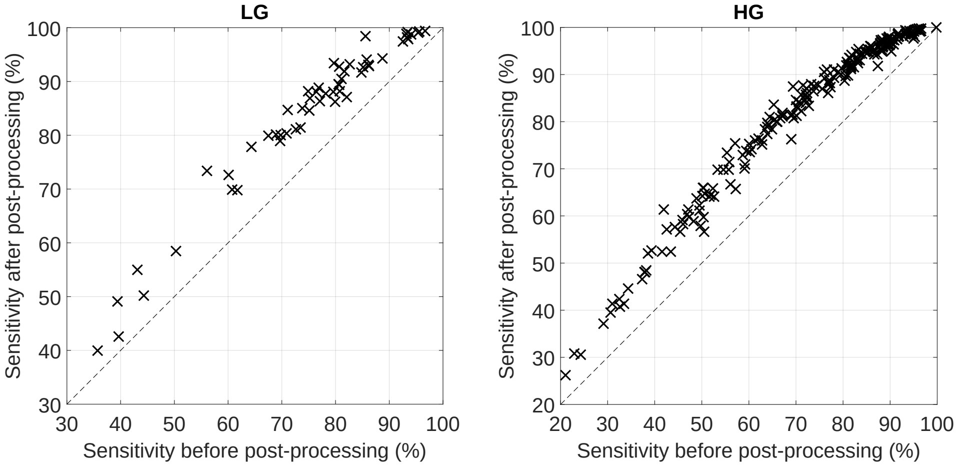

Figure 11: TPR values obtained for individual LG (left panel) and HG (right panel) tumor volumes, plotted after post-processing vs. before post-processing.

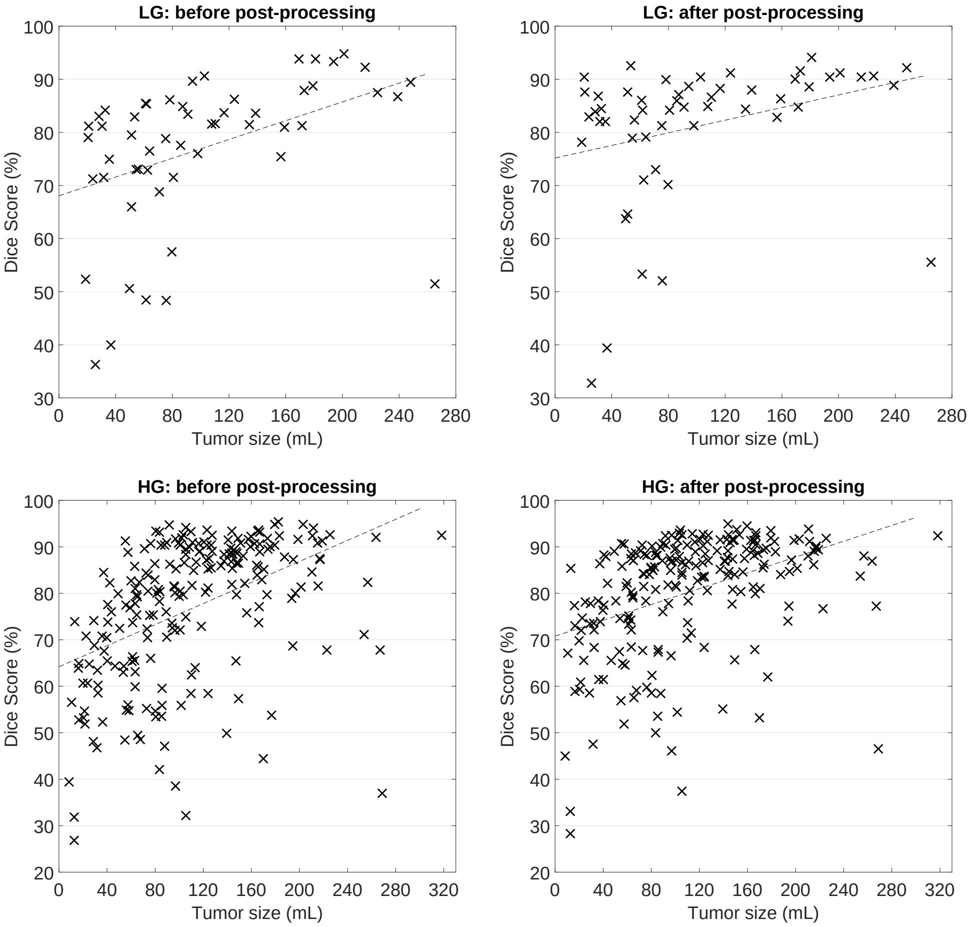

Figure 12: Dice score values obtained for individual LG (top row) and HG (bottom row) tumor volumes separately, represented against tumor size. The linear trend was also identified and indicated by the dashed lines. Left panel shows the benchmarks of the random forest's output, while the right panel the benchmarks of the post-processed segmentation.

The segmentation of a single volume ranges between 60 and 75 seconds, when executed on a single core of a PC with i7 processor running at 3.4 GHz frequency, which can be reduced below 20 seconds when executed in parallelized version on four cores. The largest computational burden represents the

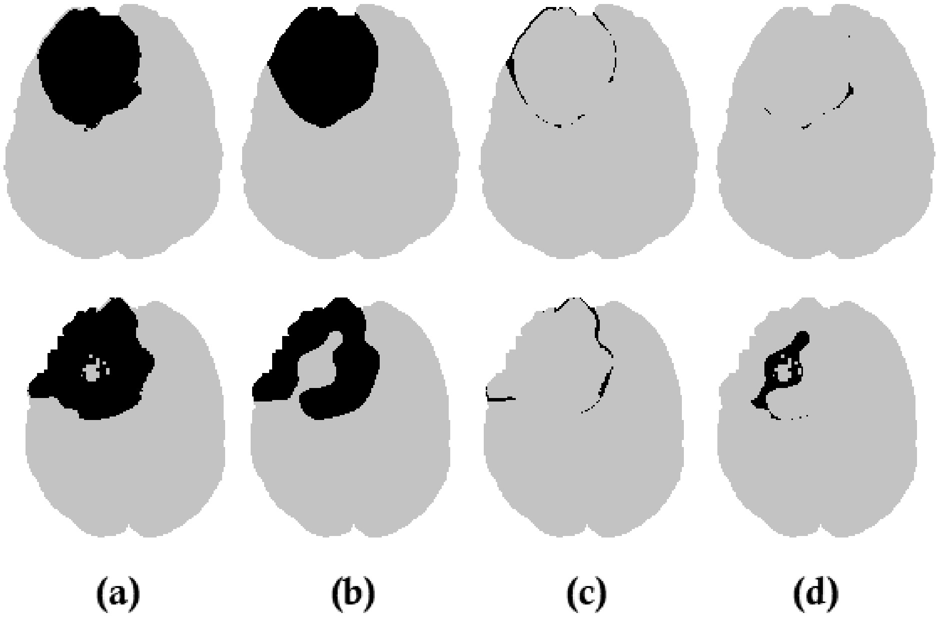

Figure 13: Detailed results presented for two slices: (a) the actual tumor (ground truth) is shown in black; (b) the segmented tumor; (c) false positives; (d) false negatives.

extraction of the 100 extra features for the approximately 1.5 million voxels of the volume.

The overall Dice score over 83.5% allows us to detect the presence of the tumor in a great majority of cases. However, the accuracy indicators can be further improved the following ways:

1. Using further texture features extracted from the neighborhood of each voxel.
2. Employing an effective feature selection scheme to eliminate useless features.
3. Implementing a more complex post-processing that investigates the contiguous ensembles of detected tumor voxels and discard small ones.

An objective comparison with existing methods enumerated in the Introduction is not an easily accomplishable task, as not all of them used the BRATS data set for evaluation, and even those which did, they did not evaluate all

Feature name			Rate of usage
Data channel	Operation	Neighborhood	
T1	Median	11×11	99.38%
FLAIR	Average	11×11	94.66%
FLAIR	Average	9×9	53.30%
FLAIR	Maximum	$3 \times 3 \times 3$	32.53%
T2	Average	11×11	26.24%
T1c	Average	11×11	20.63%
FLAIR	Average	$3 \times 3 \times 3$	19.43%
T2	Maximum	$3 \times 3 \times 3$	18.35%
FLAIR	Average	7×7	18.34%
T1c	Median	11×11	17.00%
FLAIR	Median	3×3	15.85%
FLAIR	Median	11×11	14.45%
FLAIR	Average	5×5	12.17%
T2	Minimum	$3 \times 3 \times 3$	11.77%
T1	Median	11×11	10.75%
FLAIR	Median	5×5	10.39%

Table 4: Most frequently used features

the 54 available low-grade and/or 220 available high-grade tumor volumes. With respect to the methods involved in the comparison in [21], our proposed methodology seems competitive, and it will further improve with the implementation of the above listed ideas.

To optimize the efficiency of the random forest classifier, it is useful to know how many times the features are used in decision making during the testing phase, which are the most and least used features? The number of trees in the random forest was denoted by n_T. Let us denote the number of test voxels by N_{test}. Performing the whole classification requires to perform $n_T \times N_{test}$ tests on binary decision trees. Let us further suppose, that feature number f is used at least once in N_f tests, with $0 \leq N_f \leq n_T \times N_{test}$. Having all these assumed, the rate of usage of feature number f is $N_f/(n_T \times N_{test})$. Table 4 lists those features, whose rate of usage exceeds 10%, in decreasing order of the rate of usage. There are three important things to remark:

1. Data channels FLAIR and T2 are apparently more useful than the other two.
2. The four observed features do not appear in the table, indicating their reduced usefulness in the decision making. However, they cannot be called useless, because all other computed features are extracted from them.
3. Eliminating the least used features from the system may reduce the computation burden without damaging the segmentation accuracy.

4 Conclusion

This paper presented an automatic tumor detection and segmentation algorithm employing random forests of binary decision trees. The proposed methodology reliably detects both LG and HG tumors if their volume exceeds $10\,\mathsf{mL}$. It is likely to obtain finer segmentation accuracy in the future via implementing some of the above mentioned further development ideas. We will also concentrate on differentiating among the parts of the whole tumor (edema, tumor core, necrosis, active tumor), according to the grand truth provided by the BRATS database.

Acknowledgements

This research was funded by the Sapientia Institute for Research Programs (KPI). The work of L. Szilágyi was additionally funded by the János Bolyai Research Fellowship program of the Hungarian Academy of Sciences. The work of Z. Kapás was additionally funded by the Szekely Forerunner Federation. The work of Zs. Szabó was additionally funded by the Accenture Student Research Scholarship program.

References

[1] S. B. Akers, Binary decision diagrams, *IEEE Trans. Computers* **C-27,** 6 (1978) 509–516. ⇒115

[2] A. J. Asman, B. A. Landman, Out-of-atlas labeling: a multi-atlas approach to cancer segmentation, *Proc. IEEE International Symposium on Biomedical Imaging*, Barcelona, Catalunya, 2012, pp. 1236–1239. ⇒111

[3] L. Breiman, Random forests, *Machine Learning* **45,** 1 (2001) 5–32. ⇒117

[4] J. D. Christensen, Normalization of brain magnetic resonance images using histogram even-order derivative analysis, *Magn. Reson. Imaging* **21,** 7 (2003) 817–820. ⇒114

[5] S. Ghanavati, J. Li, T. Liu, P. S. Babyn, W. Doda, G. Lampropoulos, Automatic brain tumor detection in magnetic resonance images, *Proc. IEEE International Symposium on Biomedical Imaging*, Barcelona, Catalunya, 2012, pp. 574–577. ⇒111

[6] N. Gordillo, E. Montseny, P. Sobrevilla, State of the art survey on MRI brain tumor segmentation, *Magn. Reson. Imaging* **31** (2013) 1426–1438. ⇒111, 112

[7] A. Hamamci, N. Kucuk, K. Karamam, K. Engin, G. Unal, Tumor-Cut: segmentation of brain tumors on contranst enhanced MR images for radiosurgery applications, *IEEE Trans. Med. Imaging* **31** (2012) 790–804. ⇒111

[8] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P. M. Jodoin, H. Larochelle, Brain tumor segmentation with deep neural networks, *Med. Image Anal.* **35** (2017) 18–31. ⇒112

[9] M. Y. Huang, W. Yang, Y. Wu, J. Jiang, W. F. Chen, Q. J. Feng, Brain tumor segmentation based on local independent projection-based classification, *IEEE Trans. Biomed. Eng.* **61** (2014) 2633–2645. ⇒112

[10] J. E. Iglesias, M. R. Sabuncu, Multi-atlas segmentation of biomedical images: A survey, *Med. Image Anal.* **24** (2015) 205–219. ⇒112

[11] A. Islam, S. M. S. Reza, K. M. Iftekharuddin, Multifractal texture estimation for detection and segmentation of brain tumors, *IEEE Trans. Biomed. Eng.* **60** (2013) 3204–3215. ⇒112

[12] J. Juan-Albarracín, E. Fuster-Garcia, J. V. Manjón, M. Robles, F. Aparici, L. Martí-Bonmatí, J. M. García-Gómez, Automated glioblastoma segmentation based on a multiparametric structured unsupervised classification, *PLoS ONE* **10** 5 (2015) e0125143. ⇒112

[13] V. G. Kanas, E. I. Zacharaki, C. Davatzikos, K. N. Sgarbas, V. Megalooikonomou, A low cost approach for brain tumor segmentation based on intensity modeling and 3D random walker, *Biomed. Sign. Proc. Control* **22** (2015) 19–30. ⇒112

[14] Z. Kapás, L. Lefkovits, D. Iclănzan, Á. Győrfi, B. L. Iantovics, Sz. Lefkovits, S. M.. Szilágyi, L. Szilágyi, Automatic brain tumor segmentation in multispectral MRI volumes using a random forest approach, *Proc. Pacific-Rim Symposium on Image and Video Technology (PSIVT'17)*, *Lecture Notes in Artificial Intelligence* **10749** (2018) 137–149. ⇒112

[15] M. Lê, H. Delingette, J. Kalpathy-Cramer, E. R. Gerstner, T. Batchelor, J. Unkelbach, N. Ayache, Personalized radiotherapy planning based on a computational tumor growth model, *IEEE Trans. Med. Imaging* **36** (2017) 815–825. ⇒112

[16] B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, et al., The multimodal brain tumor image segmentation benchmark (BRATS), *IEEE Trans. Med. Imaging* **34,** 10 (2015) 1993–2024. ⇒114, 118

[17] B. H. Menze, K. van Leemput, D. Lashkari, T. Riklin-Raviv, E. Geremia, E. Alberts, et al. , A generative probabilistic model and discriminative extensions for brain lesion segmentation – with application to tumor and stroke, *IEEE Trans. Med. Imaging* **35** (2016) 933–946. ⇒112

[18] I. Njeh, L. Sallemi, I. Ben Ayed, K. Chtourou, S. Lehericy, D. Galanaud, A. Ben Hamida, 3D multimodal MRI brain glioma tumor and edema segmentation: a graph cut distribution matching approach, *Comput. Med. Image Anal.* **40** (2015) 108–119. ⇒111

[19] L. G. Nyúl, J. K. Udupa, X. Zhang, New variants of a method of MRI scale standardization, *IEEE Trans. Med. Imaging* **19,** 2 (2010) 143–150. ⇒111, 114

[20] S. Pereira, A. Pinto, V. Alves, C. A. Silva, Brain tumor segmentation using convolutional neural networks in MRI images, *IEEE Trans. Med. Imaging* **35** (2016) 1240–1251. ⇒112

[21] A. Pinto, S. Pereira, D. Rasteiro, C. A. Silva, Hierarchical brain tumour segmentation using extremely randomized trees, *Patt. Recogn.* **82** (2018) 105–117. ⇒112, 129

[22] J. Sahdeva, V. Kumar, I. Gupta, N. Khandelwal, C. K. Ahuja, A novel content-based active countour model for brain tumor segmentation, *Magn. Reson. Imaging* **30** (2012) 694–715. ⇒111

[23] H. C. Shin, H. R. Roth, M. C. Gao, L. Lu, Z. Y. Xu, I. Nogues, J. H. Yao, D. Mollura, R. M. Summers, Deep nonvolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning, *IEEE Trans. Med. Imaging* **35** (2016) 1285–1298. ⇒112

[24] Zs. Szabó, Z. Kapás, Á. Győrfi, L. Lefkovits, S. M. Szilágyi, L. Szilágyi, Automatic segmentation of low-grade brain tumor using a random forest classifier and Gabor features, *Proc. 14th International Conference on Fuzzy Systems and Knowledge Discovery*, Huangshan, China, 2018, pp. 1106–1113. ⇒112

[25] L. Szilágyi, L. Lefkovits, B. Benyó, Automatic Brain Tumor Segmentation in multispectral MRI volumes using a fuzzy c-means cascade algorithm, *Proc. 11th International Conference on Fuzzy Systems and Knowledge Discovery*, Zhangjiajie, China, 2015, pp. 285–291. ⇒112

[26] N. J. Tustison, K. L. Shrinidhi, M. Wintermark, C. R. Durst, B. M. Kandel, J. C. Gee, M. C. Grossman, B. B. Avants, Optimal symmetric multimodal templates and concatenated random forests for supervised brain tumor segmentation (simplified) with ANTsR, *Neuroinformatics* **13** (2015) 209–225. ⇒112

[27] U. Vovk, F. Pernuš, B. Likar, A review of methods for correction of intensity inhomogeneity in MRI, *IEEE Trans. Med. Imaging* **26** (2007) 405–421. ⇒111

[28] R. Zaouche, A. Belaid, S. Aloui, B. Solaiman, L. Lecornu, D. Ben Salem, S. Tliba, Semi-automatic method for low-grade gliomas segmentation in magnetic resonance imaging, *IRBM* **39** (2018) 116–128. ⇒112

[29] N. Zhang, S. Ruan, S. Lebonvallet, Q. Liao, Y. Zhou, Kernel feature selection to fuse multi-spectral MRI images for brain tumor segmentation, *Comput. Vis. Image Undestand.* **115** (2011) 256–269. ⇒112

Received: Aug. 5, 2018 • Revised: Aug. 15, 2018