2014 22nd International Conference on Pattern Recognition

Efficient interactive brain tumor segmentation as within-brain kNN classification

Mohammad Havaei[1] Pierre-Marc Jodoin[1,2] Hugo Larochelle[1]

[1]Université de Sherbrooke
2500 boul. de l'Université
Sherbrooke, J1K 2R1, Canada

[2]Imeka inc.
3000 boul. de l'Université
Sherbrooke, J1K 0A5, Canada

***Abstract*—We consider the problem of brain tumor segmentation from magnetic resonance (MR) images. This task is most frequently tackled using machine learning methods that generalize *across brains*, by learning from training brain images in order to generalize to novel test brains. However this approach faces many obstacles that threaten its performance, such as the ability to properly perform multi-brain registration or brain-atlas alignment, or to extract appropriate high-dimensional features that support good generalization. These operations are both non-trivial and time-consuming, limiting the practicality of these approaches in a clinical context. In this paper, we propose to side step these issues by approaching the problem as one of *within brain* generalization. Specifically, we propose a semi-automatic method that segments a given brain by training and generalizing within that brain only, based on some minimum user interaction. We investigate how k nearest neighbors (kNN), arguably the simplest machine learning method available, combined with the simplest feature vector possible (raw MR signal + (x,y,z) position) can be combined into a method that is both simple, accurate and fast. Results obtained on the online BRATS dataset reveal that our method is fast and second best in terms of the complete and core test set tumor segmentation.**

I. Introduction

It is estimated that worldwide, more than 12 million new cancer cases occur every year and more than 7 million people will die from it (that is about 21,000 cancer deaths a day) [1]. Brain cancer is among the most aggressive ones with a survival rate after 5 years of less than 25%. Statistics are even more severe when considering malignant glioblastomas whose survival rate rarely exceeds more than 20 months. With devastating effects on the quality of life and potential cognitive impairments, often the only solution is to remove the tumor with an operation.

Magnetic resonance imaging (MRI) is the most widely used modality for brain tumor analysis. In current practice, using an MRI and sometimes with the help of a biopsy, a doctor would diagnose a tumor to be benign or malignant and can follow its evolution in time as well as the treatment that goes with it. The best way of doing so is by measuring the size and the shape of the tumor with a segmentation method. But segmentation is often done entirely manually, which is a very tedious and time consuming task [1].

Many methods have been proposed to simplify brain tumor segmentation. While some methods rely on human intervention, others are fully automatic. Automatic methods (which often use a machine learning approach) are quite popular and have been proven to be efficient, even on challenging data [1]. The advantage of these automatic methods is that once training is done, any new brain is processed without human intervention. However, these methods suffer from important limitations. First, these methods train on multiple brain subjects, for which data might have been gathered from different MRI scanners. This is problematic since the intensities of MR images are not standardized across MRI scanners. Also, manually segmented brain images for training are not so frequent and require lots of man power to build. Other obstacles include inter-slice intensity variations, differences in the noise produced by different MRI scanners, tumors-related problems when registering and aligning images, etc. To overcome the above mentioned problems, many pre-processing steps are needed, which usually require a lot of tuning effort [2]. Also, some methods rely on features that are so high-dimensional that they can hardly be stored in memory [3]. As for semi-automatic methods go, they often require a lot of user intervention and usually do not perform well [1].

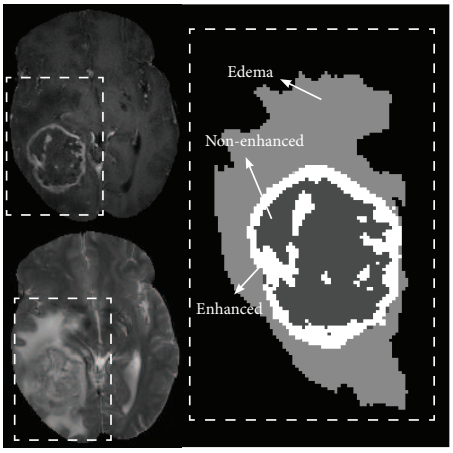

Fig. 1. Illustration of the brain segmentation task from MR images. **Left:** T1c and T2 modality images. **Right:** ground truth tumor segmentation (BRATS2013 dataset) [4].

In this paper, we consider the specific problem of segmenting an imaged brain into 4 classes: edema, non-enhancing tumor, enhancing tumor and healthy tissue (see Fig. 1). Note that the non-enhancing tumor sometimes include necrotic tissue. Our approach is halfway between automatic and semi-automatic methods. While machine learning methods train on a pre-selected set of brains and then generalize to testing brains, our method implements a "single brain" supervised learning method. The user roughly selects brain voxels associated to each class and then these voxels are used as training data. The method then generalizes by labeling non-selected voxels.

[1]www.cancer.org

1051-4651/14 $31.00 © 2014 IEEE
DOI 10.1109/ICPR.2014.106

The main characteristics of our method are as follows:

- Since it treats each brain as a separate dataset, it is immune to the multi-MRI disadvantages mentioned above.
- Although it uses only 6 simple features and relies on kNN (the simplest machine learning approach available) it produces highly accurate results.
- The entire process for a $240 \times 240 \times 168$ brain takes approximately 1 minutes on Matlab which is much faster than most state-of-the-art methods.

II. Related Work

Brain tumor segmentation methods can be divided in two great families : interactive (or semi-automatic) methods and automatic methods.

Interactive methods are those relying on user intervention. Many of these methods rely on active deformable models (*e.g.* snakes). For these methods, the user initializes a contour around the region of interest, *i.e.* the tumor. The active contour then converges slowly to its closest optimal configuration. It is assumed that the global minimum energy is achieved when the contour reaches the borders of the tumor. Jiang [5] uses a level set method to perform tumor segmentation. Wang [6] proposed the fluid vector flow active contour model that improves its capture range in MR images. Efforts have been made to initialize the contour automatically and therefore eliminate the need for human interaction. Ho [7] used the difference between pre and post contrast T1 together with a mixture of Gaussians (GMM) to compute a probability map for the tumor, which can be used to automatically initialize the active contour. Rexilious [8] initializes the segmentation by a tumor probability map based on global cross subject intensity variability, which is achieved by histogram matching. One problem with deformable models is that they are highly dependent on the image gradients and if the tumor region does not have well defined borders, they are likely to fail. Also, strong gradients from surrounding objects may attract the active contour in the wrong direction. Moreover, it is not trivial to integrate multiple MRI modalities into these algorithms. Also, since snakes and level set methods are fundamentally 2-class segmentation methods, it is non trivial to make it segment $N > 2$ classes as is often required for tumor segmentation.

As for automatic methods, they are often based on machine learning classification techniques [1]. The choice of features can then play a crucial role in the ability of the method to generalize well. For example, Jenson [9] combined morphological, diffusion weighted and perfusion weighted features to train a two hidden layer neural network classifier across patient brains. Other methods have used random forests for classification. Reza [10] used T1, T2 and FLAIR along with other intensity and texture features. They trained a classical random forest model to perform classification on this feature space. Festa [3] used series of intensity based features, texture features and neighborhood information features. A total of 300 features were computed. Classical decision forest comprising of 50 trees is trained on this feature space. Tustison [11] constructed a large feature space using first order neighborhood statistical images, GMM and Markov Random Field (MRF) posteriors, and template differences. The advantage of machine learning classification methods is that it is possible to integrate many features, even if they are redundant. The drawback is that they can be vulnerable to overfitting, which is a likely possibility when generalizing across different brain images.

Classification methods can be further improved by modeling the spatial dependencies between voxel labels, typically with an MRF or a Conditional Random Field (CRF). Lee [12] performed 2-class segmentation (tumor vs. non-tumor) using an SVM and a variation of conditional random fields to account for neighborhood relationships. Nie [13] proposed an automatic method using a GMM, where Iterated Conditional Modes (ICM) is used to incorporate spatial relationships. Bauer [14] used a kernel SVM for multiclass segmentation of brain tumors, where a CRF is used to regularise the results. Meier [2] used density forests and classification forests in a generative-discriminative framework. Having found the posterior probabilities, they used a CRF to incorporate similarity between the probability distribution of neighboring voxels. Zhao [15] proposed a method based on histogram matching. Using T1c, T2 and FLAIR modalities they built a 3D joint histogram. At test time, the likelihood for every voxel is computed by registering the test brain to every training brain and performing histogram matching. The likelihood is later used in a CRF to perform segmentation.

While our method falls squarely within semi-automatic methods, it shares with automatic methods the use of a machine learning classification algorithm, ran on a feature representation of voxels and improved by a spatial dependency model. The main difference is that generalization is performed *within* each brain, based on the training data provided by the user's interaction. This simplified generalization problem allows us to use a very simple feature space, yielding an interactive segmentation method that is very fast and effective. Vaidyanathan [16] used a similar, semi-automatic, kNN classification mehtod, applied to proton density, T1 and T2 modalities. Cai [17] also proposed a semi-automatic segmentation method that uses instead Quadratic Discriminative Aanalysis to perform multi-class segmentation. However, they did not use the (x,y,z) voxel positions as features (see Section III-B) nor did they deal with label spatial dependency modeling (see Section III-C), which we've found to both play a crucial role in obtaining competitive performances.

III. Our method

In this section, we describe the different components of our approach. Specifically, we explain the voxel feature representation we used, how kNN classification was performed using the user interaction labeled data and which spatial label dependency models we investigated.

A. Feature representation and manual selection

We considered the use of 3 MRI modalities, known as contrasted T1 (T1 with gadolinium), T2 and FLAIR (extension to a different set of modalities is trivial). These are co-registered and stored as separate channels in a 3$^{\text{rd}}$-order tensor image I, where each voxel in I is a 3D vector containing all 3 modalities. Let I_v^1, I_v^2, I_v^3 be the 3 modalities for a given voxel v. Our goal is to use a machine learning classifier to predict

the voxel label T_v from a feature representation of voxel v. We thus need to convert each such voxel v into an N-dimensional feature representation F_v.

We propose to use a very simple feature representation, consisting in all 3 values of v under the MRI modalities as well as the 3D relative position of v: $F_v = (I_v^1, I_v^2, I_v^3, x, y, z)$. These features are normalized between 0 and 1.

To collect voxel label data for a given brain image to segment, the first step of our method is then for the user to roughly select a subset of voxels associated with each class. This is done through a graphical interface. We will note as B the binary mask such that $B_v \in \{0, 1\}$ indicates whether a voxel v has been manually selected (*i.e.* labeled) or not. T will then be the class-selection mask where $T_v \in \{$edema, non-enhancing tumor, enhancing tumor, healthy$\}$ is the class label associated with the voxel v by the user.

At this point, from each selected voxel, we can generate a training pair (F_v, T_v) and construct a training set $\mathcal{D}$ that we shall use to classify the non-selected voxels using a classifier.

B. Voxel classifier

Once manual selection is done and each voxel has been assigned a 6D feature vector, the goal is to generalize to non-selected voxels, that is to classify those voxels for which $B_v = 0$. One of the simplest classifier is arguably the k nearest neighbor (kNN) classifier. For each non-selected voxel v, we first consider $\mathcal{N}_v$, the set of k nearest neighbors of F_v among the training voxels, irrespective to their class label. Let $\mathcal{N}_v = ((F_{v1}, T_{v1}), (F_{v2}, T_{v2}), ..., (F_{vk}, T_{vk}))$ where F_{vi} is the i^{th} closest training point of F_v. The kNN classification rule assigns a class label to some voxel v following this equation

$$T_v = \arg\max_c \frac{1}{k} \sum_{(F_{vi}, T_{vi}) \in \mathcal{N}_v} \delta(T_{vi}, c) \tag{1}$$

where c is a class label and $\delta(a, b)$ returns 1 when $a = b$ and 0 otherwise. We refer to the segmentation method classifying each voxel independently in this way as simply **kNN**.

As stated earlier in Section III-A, we rely on a simple 6D voxel feature representation to perform this classification. This has the important advantage of allowing for very efficient predictions, compared to using hundreds of features. In contrast to automatic methods, this might seem like an over-simplistic feature representation. However, the use of a simplified feature set is made possible by the fact that we only attempt to generalize *within* a given brain, as opposed to *across* a different brain image. Indeed, this later type of generalization is much harder, as it requires being robust to variations related to changes in MRI scanner or registration/alignment procedures, unlike in *within* brain generalization.

Relying on *within* brain generalization is also what allows us to use the voxel 3D relative positions as features (these would obviously not be useful across brains). This is probably why spatial coordinate features are rarely used in other systems. Yet, we've found these features to play a crucial role in obtaining good segmentation performances. They allow segmented regions to be locally grounded around the user-labeled voxels and not produce false positives far away from the tumor.

C. Spatial label dependency models

As mentioned earlier, segmentation accuracy can easily be improved by leveraging a model of the 3D spatial regularity of labels. We describe here two such models that we considered in our experiments.

1) Markov Random Fields (MRF): One way of enforcing spacial regularity is through an MRF formulation. Let F be the set of features F_v and T be the set of labels T_v, for all voxels v. A common MRF model for the joint distribution $P(F, T)$ is one that decomposes as follows:

$$\begin{aligned} p(F,T) &= P(F|T)P(T) = P(T)\prod_v P(F_v|T_v) \\ &= \frac{1}{Z}\prod_v \exp\left(V(F_v,T_v) + \sum_{r\in\eta_v} W(T_v,T_r)\right) \end{aligned} \tag{2}$$

where η_v corresponds to the 6 voxel spatial neighbors of v in 3D, Z is a normalization constant, and $V(F_v, T_v)$ and $W(T_v, T_r)$ are likelihood and prior energy functions. One can associate with this distribution a graph in which the labels T_v are organized into a 3D grid where each T_v is connected to its neighbors η_v as well as to its associated feature vector node F_v. A property of this MRF is then that it satisfies the local Markov property given by that graph, whereby a node is independent of all others given its neighbors [18].

The role of the energy functions $V(F_v, T_v)$ and $W(T_v, T_r)$ is to express preferences for the assignment of the variables in the graph. Following section III-B, a good choice for deriving our preferences for the values taken by a pair (F_v, T_v) is our kNN classifier, which can serve as a simple but efficient likelihood energy function:

$$V(F_v, T_v) = -\log\left(\frac{1}{k}\sum_{(F_{vi}, T_{vi}) \in \mathcal{N}_v} \delta(T_{vi}, T_v)\right) \tag{3}$$

where $\mathcal{N}_v$ contains the k nearest neighbors of F_v (gathered within the training dataset $\mathcal{D}$). As for $W(T_v, T_r)$, we use a Potts model which returns 0 when $T_v = T_r$ and β (a hyper-paramter) otherwise. This can be formulated as

$$W(T_v, T_r) = \beta(1 - \delta(T_v, T_r)) \tag{4}$$

Segmenting a brain image can then be formulated as the problem of finding the most likely value for all labels T_v that have not be selected and labeled by the user, given the voxel features. Noting $\mathcal{T}$ as the set of label assignments that are consistent with the user labeling, segmentaiton is thus equivalent to

$$\begin{aligned} T &= \arg\max_{T\in\mathcal{T}} P(T|F) = \arg\max_{T\in\mathcal{T}} P(F,T) \\ &= \arg\min_{T\in\mathcal{T}} \sum_v \left(V(F_v,T_v) + \sum_{r\in\eta_v} W(T_v,T_r)\right). \end{aligned} \tag{5}$$

To minimize equation (5), we use graph-cut and the alpha-expansion algorithm [19]. In our experiments, we refer to this segmentation method using the MRF label dependency model as **kNN-MRF**.

2) Conditional Random Fields (CRF): The MRF model of equation (2) defines a joint distribution $P(F,T)$, which makes certain simplifying assumptions about the distribution over F, such as the conditional independence of each F_v given the value of T_v. One consequence of this choice is that neighboring voxel features $F_{v'}$ of a label T_v (where $v' \in \eta_v$) cannot directly express a preference on the value of T_v. This yields a model for the posterior $P(T|F)$ that is too weak in practice to model the complex structures found in medical images.

Conditional Random Fields (CRF) is an alternative to MRF which models directly the posterior probabilities of the labels given the features $P(T|F)$. This alleviates the need to model the distribution over the feature vectors F and allows us to construct a richer conditional $P(T|F)$. For our CRF model, we use

$$T = \arg\min_{T\in\mathcal{T}} \sum_v \left(V(F_v, T_v) + \sum_{r\in\eta_v} I(T_v, F_v, T_r, F_r) \right). \tag{6}$$

The unary term, $V(F_v, T_v)$ is computed the same as in (3) and the pairwise term is defined as

$$I(T_v, F_v, T_r, F_r) = \lambda(1 - \delta(T_v, T_r)) \exp\left(\frac{-\|F_v - F_r\|}{\sigma^2} \right). \tag{7}$$

We refer to the segmentation method using this label dependency model as **kNN-CRF**.

IV. Experimental Results

A. Setup

The dataset used for this research was provided by the BRain Tumor Segmentation (BRATS) challenge as part of the MICCAI-2013 conference. The challenge contains training and test sections. For the training part, a total of 30 real patient subjects (20 with high-grade tumors and 10 with low-grade) and 30 simulated subjects along with the corresponding ground truths are provided. The test set contains 10 real patient brains with high-grade tumors.

Although BRATS provides separate challenges for real patient subjects and simulated subjects, we only focused on real patient subjects. Even though ground truth labels were provided for training data, since our method does not rely on cross-subject training, the ground truth was only used for the validation of our method. Overall, the method was tested on 40 subjects (30 from training data and 10 from test data). We used the same model hyper-parameters for all subjects and through all experiments. They are as follows. For the k nearest-neighbor algorithm, $k = 3$ closest neighbors were considered. For the MRF algorithm, the smoothness term β was set to 0.2 and in the CRF algorithm σ and λ were set to 0.1 and 0.5 respectively. These values were selected by cross-validation on 6 brains in the training set.

As for the user interaction, we labeled voxels in only two 2D slices of each brain, for each class, from a graphical interface. On average, only 0.4% of the voxels containing pathology and 0.03% of the voxels corresponding to healthy tissue were manually selected, thus providing minimal labeled data to the algorithm. The same selection masks were used by kNN, kNN-MRF and kNN-CRF.

B. Results and Discussion

The quantitative results and global evaluation reported in this section come from the online BRATS system [4], which also reports the performance of other state-of-the-art methods (both automatic and semi-automatic could participate). Performance metrics include Dice, precision and recall. As mentioned on the BRATS website, these measures are computed for 3 different subsets of tumor classes, namely, complete tumor, tumor core and enhancing tumor map. Complete tumor includes edema, non-enhancing and enhancing tumor while tumor core include non-enhancing and enhancing tumor. Enhancing tumor includes only enhancing tumor.

Evaluation results for kNN , kNN-MRF and kNN-CRF along with the other registered methods, on the training and test data, are presented in Tables I and II respectively. These results are reported by the online evaluation system in BRATS website [4]. These tables show the average performance across every brain for each metric. An average ranking system is applied to rank different methods.

As far as the average ranking goes, kNN-CRF performs better than kNN and kNN-MRF. This is not very surprising since the use of a prior term helps regularizing kNN. But the fact that the pairwise label term is data-dependent allows kNN-CRF to generate resuls that better fit the edges of the images. This is shown in Fig. 2 where the results by kNN is noisier than for the other methods. Also, the borders of the kNN-CRF tumor map better fit on the edges of the T1c and T2 images than kNN-MRF.

Another observation from these tables is that contrary to several other methods, our 3 kNN methods provided similar score on both the training and testing data. For example, the complete Dice score for Reza [10] (which is first on the training dataset) goes from 0.92 on the training dataset down to 0.83 on the testing dataset. Same with Doyle [20] whose complete Dice score go from 0.83 down to 0.71. This is a clear sign that these automatic methods can be vulnerable to overfitting on the training data. This is not the case with our method since it trains on each brain independently.

On the test dataset, our kNN-CRF method ranks second on the Dice complete tumor (0.85) and Dice core tumor (0.75). This is an important conclusion since core tumors is what neurosurgeons remove during surgery. Complete and core tumor segmentation is thus of primary importance in the context of a preoperative application. We also implemented kNN as presented by Vaidyanathan [16], but since they dont use the (x,y,z) features nor any spatial regularization, the results were significantly lower than our plain kNN shown in Tables I and II.

As far as processing goes, our kNN-CRF and kNN-MRF methods take between 1 and 2 minutes to process a brain on Matlab. This is significantly faster than many other methods. For example, Meier [2] report that their method takes up to 15 minutes per brain. Let us mention that a preliminary GPU implementation makes us believe that a GPU version of kNN and graph cut should take processing time down to a few seconds.

As explained earlier, part of the reason why our method is fast is that it relies on a small set of easily extracted features.

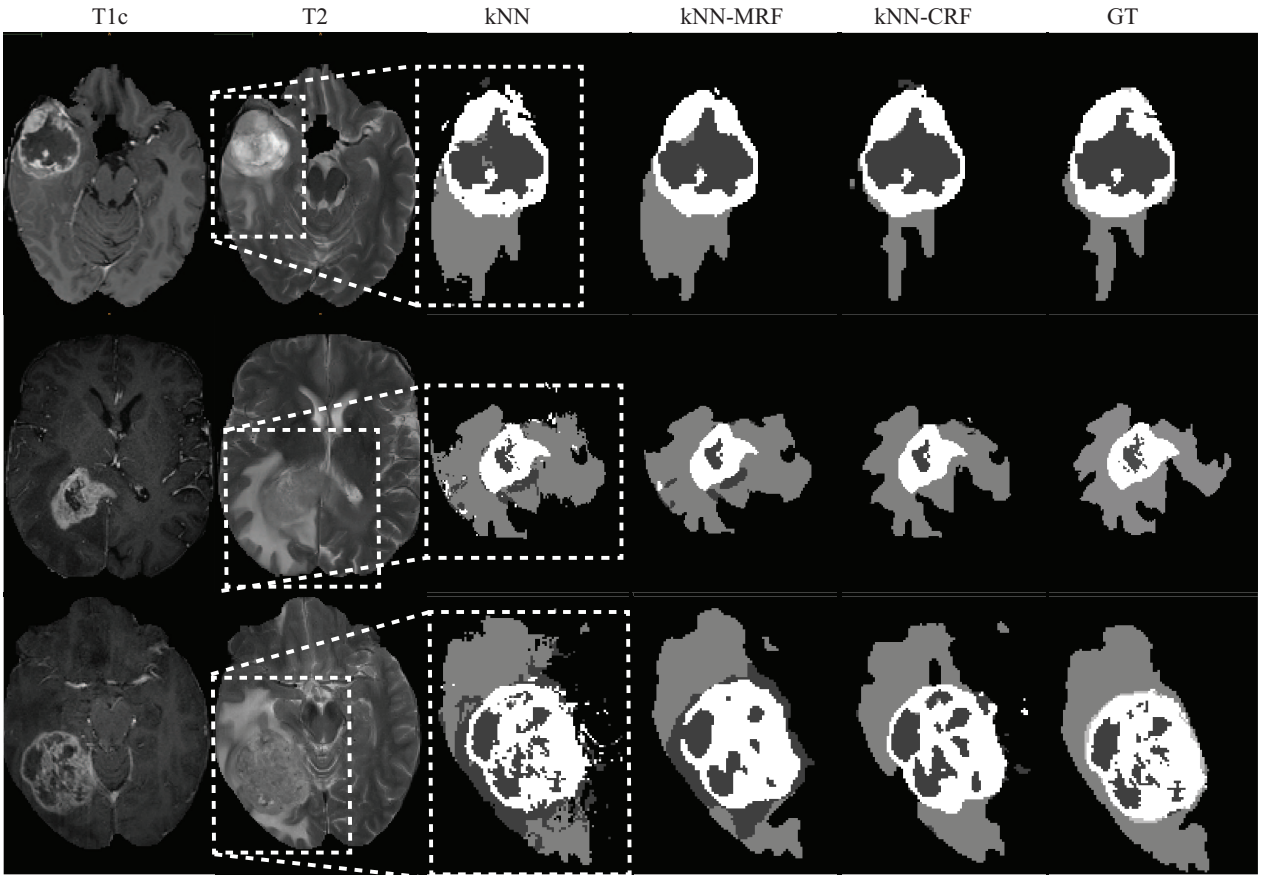

Fig. 2. Illustration of brain tumor segmentation maps predicted by our method, for four different brains (rows). From left to right, the columns show the T1c modality, the T2 modality, the segmentations predicted by kNN, kNN-MRF and kNN-CRF, and the ground truth segmentation.

Furthermore, the fact that we use a 6D features makes our method memory efficient. Since, brain images contain about $240 \times 240 \times 168$ voxels, methods using high dimensional feature vectors (Festa [3] compute up to 300 features per voxel) will require gigabytes of memory to store one single brain.

Top ranking methods on the test set [11], [2] use random forests applied on a larger set of features. However, computing many features is time consuming. Tustison [11] reported that feature extraction takes up to 11.5 hours for 30 subjects (24 minutes per subject). This is due to the fact that their features require cross-subject registration. Doyle [20] also reported a processing time of 30 minutes per patient.

According to the BRATS challenge, our method is the top performing semi-automatic method. The only other semi-automatic method is the one by Guo [21], whose overall score is significantly lower than ours.

V. Conclusion

In this paper, we presented a simple, fast, and memory efficient method for brain tumor segmentation. Instead of training on a given set of brains and generalizing on other brains, our method trains and generalizes on each brain independently. This is done with minimal user intervention, which roughly selects tumorous regions in only 2 slices.

Our kNN-CRF method is globally ranked 3rd on the training and testing datasets and provides the second best results on complete and core tumors. Our method is also very fast compared to other methods with approximately 1 minute per brain on a Matlab implementation. We intend to implement our method on a GPU, which should bring down processing time to a few seconds.

In future work, we would like to further develop the idea of *within* brain generalization by considering the use of machine learning classifiers other than kNN, such as kernel SVMs, and hopefully improve the accuracy even further. Also, we would like to investigate a fully automatic variant of our approach, where the initially selected tumours regions would be automatically select, in order to remove manual interaction.

Acknowledgment

The authors would like Maxime Descoteaux, Jean-Christophe Houde, and Michael Bernier for their support with brain software and medical image formats.

References

[1] S. Bauer, R. Wiest, L. Nolte, and M. Reyes, "A survey of mri-based medical image analysis for brain tumor studies," *Physics in medicine and biology*, vol. 58, no. 13, p. R97, 2013.

Method	Description	Dice			Precision			Recall		
		Complete	Core	Enhancing	Complete	Core	Enhancing	Complete	Core	Enhancing
Reza[10]	Random forest	0.92	0.91	0.68	0.97	0.95	0.70	0.89	0.88	0.66
Zikic[4]	Random forest	0.92	0.93	0.63	0.86	0.92	0.63	0.98	0.94	0.64
Tustison[11]	RF	0.88	0.76	0.55	0.88	0.80	0.65	0.89	079	0.53
kNN-CRF	kNN+CRF	0.87	0.80	0.53	0.92	0.82	0.57	0.82	0.80	0.53
Zhao[15]	Hsit. Matching	0.86	0.75	0.55	0.88	0.86	0.65	0.86	0.72	0.50
Festa[3]	RF	0.79	0.62	0.57	0.82	0.80	0.60	0.80	0.57	0.57
Meier[2]	RF+CRF	0.80	0.66	0.53	0.78	0.72	0.56	0.85	0.67	0.51
kNN-MRF	kNN+MRF	0.84	0.74	0.51	0.83	0.71	0.47	0.87	0.82	0.61
Doyle[20]	HMF+EM	0.83	0.54	0.48	0.84	0.47	0.51	0.86	0.77	0.51
Subbanna[4]	-	0.78	0.66	0.46	0.66	0.87	0.62	0.97	0.56	0.40
kNN	kNN	0.84	0.73	0.51	0.79	0.68	0.46	0.90	0.83	0.62
Cordier[4]	Patch similarity	0.79	0.66	0.48	0.90	0.73	0.52	0.73	0.66	0.49
Taylor[4]	map-reduce HMM	0.68	0.62	0.52	0.78	0.83	0.60	0.62	0.53	0.50
Guo[21]	active contour	0.55	0.55	0.47	0.61	0.60	0.49	0.51	0.51	0.49
Buendia[4]	GAIN	0.49	0.41	0.42	0.57	0.45	0.45	0.44	0.40	0.44

TABLE I. THE AVERAGE PERFORMANCE ON THE TRAINING DATA FROM THE BRATS CHALLENGE [4], AS OF 12/12/2013.

Method	Description	Dice			Precision			Recall		
		Complete	Core	Enhancing	Complete	Core	Enhancing	Complete	Core	Enhancing
Tustison[11]	RF	0.87	0.78	0.74	0.85	0.74	0.69	0.89	0.88	0.83
Meier[2]	RF+CRF	0.82	0.73	0.69	0.76	0.78	0.71	0.92	0.72	0.73
kNN-CRF	kNN+CRF	0.85	0.75	0.63	0.92	0.84	0.77	0.80	0.73	0.56
Reza[10]	RF	0.83	0.72	0.72	0.82	0.81	0.70	0.86	0.69	0.76
Uhlich[4]	RF	0.83	0.69	0.68	0.86	0.83	0.70	0.82	0.62	0.68
Zhao[15]	Hsit. matching	0.84	0.70	0.65	0.80	0.67	0.65	0.89	0.79	0.70
kNN-MRF	kNN+MRF	0.82	0.71	0.66	0.78	0.67	0.65	0.88	0.82	0.72
kNN	kNN	0.80	0.68	0.64	0.74	0.62	0.60	0.90	0.83	0.74
Cordier[4]	Patch similarity	0.84	0.68	0.65	0.88	0.63	0.68	0.81	0.82	0.66
Festa[3]	RF	0.72	0.66	0.67	0.77	0.77	0.70	0.72	0.60	0.70
Buendia[4]	GAIN	0.70	0.57	0.65	0.68	0.53	0.65	0.74	0.65	0.72
Taylor[4]	map-reduce HMM	0.69	0.59	0.62	0.71	0.77	0.65	0.69	0.49	0.65
Doyle[20]	HMF+EM	0.71	0.46	0.52	0.66	0.38	0.58	0.87	0.70	0.55
Subbanna[4]	-	0.13	0.07	0.06	0.23	0.26	0.22	0.11	0.06	0.05

TABLE II. THE AVERAGE PERFORMANCE ON THE TEST DATA FROM THE BRATS CHALLENGE [4], AS OF 12/12/2013.

[2] R. Meier, S.Bauer, J. Slotboom, R. Wiest, and M. Reyes, "A hybrid model for moltimodal brain tumor segmentation," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 31–36, 2013.

[3] J. Festa, S. Pereira, J. Mariz, N. Sousa, and C. Silva, "Automatic brain tumor segmentation of multi-sequence mr images using random dicision forests," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 23–26, 2013.

[4] "Virtual skeleton database," http://www.virtualskeleton.ch/.

[5] C. Jiang, X. Zhang, W. Huang, and C. Meinel, "Segmentation and quantification of brain tumor," in *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2004. (VECIMS). 2004 IEEE Symposium on*, 2004, pp. 61–66.

[6] T. Wang, I. Cheng, and A. Basu, "Fluid vector flow and applications in brain tumor segmentation," *IEEE Trans. Biomedical Eng.*, vol. 56, no. 3, pp. 781–789, 2009.

[7] S. Ho, E. Bullitt, and G. Gerig, "Level-set evolution with region competition: automatic 3-d segmentation of brain tumors," in *Proc. Int. Conf. Pattern Recognition*, vol. 1, 2002, pp. 532–535.

[8] J. Rexilius, H. K. Hahn, J. Klein, M. G. Lentschig, and H.-O. Peitgen, "Multispectral brain tumor segmentation based on histogram model adaptation," in *Medical Imaging*, 2007, pp. 65 140V–65 140V.

[9] T. R. Jensen and K. M. Schmainda, "Computer-aided detection of brain tumor invasion using multiparametric mri," *Journal of Magnetic Resonance Imaging*, vol. 30, no. 3, pp. 481–489, 2009.

[10] S. Reza and K. Iftekharuddin, "Multi-class abnormal brain tissue segmentation using texture features," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 38–42, 2013.

[11] N. Tustison, M. Wintermark, C. Durst, and B. Avants, "Ants and arboles," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 47–50, 2013.

[12] C.-H. Lee, S. Wang, A. Murtha, M. R. Brown, and R. Greiner, "Segmenting brain tumors using pseudo–conditional random fields," in *Medical Image Computing and Computer-Assisted Intervention*. Springer, 2008, pp. 359–366.

[13] J. Nie, Z. Xue, T. Liu, G. S. Young, K. Setayesh, L. Guo, and S. T. Wong, "Automated brain tumor segmentation using spatial accuracy-weighted hidden markov random field," *Computerized Medical Imaging and Graphics*, vol. 33, no. 6, pp. 431–441, 2009.

[14] S. Bauer, L. Nolte, and M. Reyes, "Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization," in *Medical Image Computing and Computer-Assisted Intervention*. Springer, 2011, pp. 354–361.

[15] L. Zhao, D.Sarikaya, and J. Corso, "Automatic brain tumor segmentation with mef on supervoxels," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 51–54, 2013.

[16] M. Vaidyanathan, L. Clarke, R. Velthuizen, S. Phuphanich, A. Bensaid, L. Hall, J. Bezdek, H. Greenberg, A. Trotti, and M. Silbiger, "Comparison of supervised mri segmentation methods for tumor volume determination during therapy," *Magnetic resonance imaging*, vol. 13, no. 5, pp. 719–728, 1995.

[17] H. Cai, R. Verma, Y. Ou, S.-k. Lee, E. R. Melhem, and C. Davatzikos, "Probabilistic segmentation of brain tumors based on multi-modality magnetic resonance images," in *Biomedical Imaging: From Nano to Macro, 2007. ISBI 2007. 4th IEEE International Symposium on*. IEEE, 2007, pp. 600–603.

[18] S. Li, *Markov Random Field Modeling in Image Analysis*, 3rd ed. Springer, 2009.

[19] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 26, no. 9, pp. 1124–1137, 2004.

[20] S. Doyle, F. Vasseur, and F. Forbes, "Fully automatic brain tumor segmentation from multiple mr sequences using hidden markov fields and variational em," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 18–22, 2013.

[21] X. Guo, L.Schwartz, and B. zhao, "Semi-automatic segmentation of multimodal brain tumor using active contours," *Proc. Workshop on Brain Tumor Segmentation MICCAI.*, pp. 27–30, 2013.