Proceedings of the **Multimodal Brain Tumor Image Segmentation Challenge** held in conjunction with MICCAI 2015 (**MICCAI-BRATS 2015**)

Editors: BH Menze, M Reyes, K Farahani, J Kalpathy-Cramer, D Kwon

BACKGROUND AND INTRO

Because of their unpredictable appearance and shape, segmenting brain tumors from multi-modal imaging data is one of the most challenging tasks in medical image analysis. Although many different segmentation strategies have been proposed in the literature, it is hard to compare existing methods because the validation datasets that are used differ widely in terms of input data (structural MR contrasts; perfusion or diffusion data; ...), the type of lesion (primary or secondary tumors; solid or infiltratively growing), and the state of the disease (pre- or post-treatment).

In order to gauge the current state-of-the-art in automated brain tumor segmentation and compare between different methods, we are organizing a Multimodal Brain Tumor Image Segmentation (BRATS) challenge in conjunction with the MICCAI 2015 conference. For this purpose, we are making available a large dataset of brain tumor MR scans in which the relevant tumor structures have been delineated.

This challenge is in continuation of BRATS 2012 (Nice), BRATS 2013 (Nagoya), and BRATS 2014 (Boston).

Overall, twelve groups reported preliminary results and submitted documents describing their approaches that are collected in the following.

Bjoern Menze, Mauricio Reyes, Keyvan Farahani,
Jayashree Kalpathy-Cramer, Dongjin Kwon

Munich, August 2015

CONTENT

Brain Tumor Segmentation by a Generative Model with a Prior on Tumor Shape

Mikael Agn[1], Oula Puonti[1], Ian Law[2], Per Munck af Rosenschöld[3] and Koen Van Leemput[1,4]

[1] Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark
[2] Department of Clinical Physiology, Nuclear Medicine and PET, and
[3] Department of Oncology, Rigshospitalet, Denmark
[4] Martinos Center for Biomedical Imaging, MGH, Harvard Medical School, USA

Abstract. We present a fully automated generative method for brain tumor segmentation in multi-modal magnetic resonance images. We base the method on the type of generative model often used for healthy brain tissues, where tissues are modeled by Gaussian mixture models combined with a spatial tissue prior. We extend the basic model with a tumor prior, which uses convolutional restricted Boltzmann machines to model tumor shape. Experiments on the 2015 and 2013 BRATS data sets indicate that the method's performance is comparable to the current state of the art in the field, while being readily extendable to any number of input contrasts and not tied to any specific imaging protocol.

1 Introduction

Brain tumor segmentation from magnetic resonance (MR) images is of high value in radiosurgery and radiotherapy planning. Automatic tumor segmentation is challenging since tumor location, shape and appearance vary greatly across patients. Moreover, brain tumor images often exhibit significant intensity inhomogeneity as well as large intensity variations between subjects, particularly when they are acquired with different scanners or at different imaging facilities.

Most current state-of-the-art methods exploit the specific intensity contrast information of annotated training images, which hinders their applicability to images acquired with different imaging protocols. In this paper we propose an automated generative method that achieves segmentation accuracy comparable to the state of the art while being contrast-adaptive and readily extendable to any number of input contrasts. To achieve this, we incorporate a prior on tumor shape into an atlas-based probabilistic model for healthy tissue segmentation. The prior models tumor shape by convolutional restricted Boltzmann machines (RBMs) that are trained on expert segmentations, without the use of the *intensity information* corresponding to these segmentations.

2 Generative modeling framework

Let $\mathbf{D} = (\mathbf{d}_1, ..., \mathbf{d}_I)$ denote the multi-contrast MR data, where I is the number of voxels and $\mathbf{d}_i$ contains the intensities at voxel i. We aim to segment each voxel

i into either a healthy tissue label $l_i \in \{1, ..., K\}$ or tumor tissue $z_i \in \{0, 1\}$ and within tumor tissue into either edema or core $y_i \in \{0, 1\}$. For this purpose we build a generative model that describes the image formation and then use this model to derive a fully automated segmentation algorithm. To avoid cluttered equations we define the model in 1D; it is easily extended to the 3D images we actually use. We use the posterior of all variables given the data:

$$p(\mathbf{l}, \mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}, \boldsymbol{\theta}|\mathbf{D}) \propto p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta}) \cdot p(\mathbf{l}) \cdot p(\boldsymbol{\theta}) \cdot p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}). \tag{1}$$

The model consists of a likelihood function $p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta})$, which links labels to MR intensities, and priors $p(\mathbf{l})$, $p(\boldsymbol{\theta})$ and $p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G})$, where $\mathbf{H}$ and $\mathbf{G}$ denotes the hidden units of the RBMs (see further below). We define the likelihood as

$$p(\mathbf{D}|\mathbf{l}, \mathbf{z}, \mathbf{y}, \boldsymbol{\theta}) = \prod_i \begin{cases} p(\mathbf{d}_i|l_i, \boldsymbol{\theta}_l) & \text{if } z_i = 0 \text{ and } y_i = 0, \text{ (healthy tissue)} \\ p(\mathbf{d}_i|\boldsymbol{\theta}_e) & \text{if } z_i = 1 \text{ and } y_i = 0, \text{ (edema)} \\ p(\mathbf{d}_i|\boldsymbol{\theta}_c) & \text{if } z_i = 1 \text{ and } y_i = 1, \text{ (core)} \end{cases}, \tag{2}$$

where $\boldsymbol{\theta}$ contains the unknown model parameters $\boldsymbol{\theta}_l$, $\boldsymbol{\theta}_e$, $\boldsymbol{\theta}_c$ and bias field parameters $\mathbf{C}$ and $\boldsymbol{\phi}$; and $p(\mathbf{d}_i|l, \boldsymbol{\theta}_l) = \sum_{lg} \gamma_{lg} \mathcal{N}(\mathbf{d}_i - \mathbf{C}^T \boldsymbol{\phi}^i | \boldsymbol{\mu}_{lg}, \boldsymbol{\Sigma}_{lg})$ is a Gaussian mixture model (GMM). Subscript g denotes a Gaussian component within label l and $\mathcal{N}(\cdot)$ denotes a normal distribution; and γ_{lg}, $\boldsymbol{\mu}_{lg}$ and $\boldsymbol{\Sigma}_{lg}$ are the weight, mean and covariance of the corresponding Gaussian. The probabilities $p(\mathbf{d}_i|\boldsymbol{\theta}_e)$ and $p(\mathbf{d}_i|\boldsymbol{\theta}_c)$ are also GMMs. Furthermore, bias fields corrupting the MR scans are modeled as linear combinations of spatially smooth basis function added to the scans [4]. $\boldsymbol{\phi}^i$ contains basis functions at voxel i and $\mathbf{C} = (\mathbf{c}_1, ..., \mathbf{c}_n)$, where $\mathbf{c}_n$ denotes the parameters of the bias field model for MR contrast n.

We use a probabilistic affine atlas computed from segmented healthy subjects as the healthy tissue prior [5], defined as $p(\mathbf{l}) = \prod_i \pi_{li}$. The atlas includes probability maps of GM, WM, CSF and background (BG). Moreover, we add a prior $p(\boldsymbol{\theta})$ on the distribution parameters [6], which ensures that the Gaussians modeling tumor tissue are neither too narrow or too wide and that their mean values in FLAIR are higher than that of $\boldsymbol{\mu}_{GM}$.

Tumor prior: We model tumor shape by convolutional RBMs, which are graphical models over visible and hidden units that allow for efficient sampling over large images without a predefined size [1]. The energy term of an RBM is defined as $E(\mathbf{z}, \mathbf{H}) = -\sum_k \mathbf{h}_k \bullet (\mathbf{w}_k * \mathbf{z}) - \sum_k b_k \sum_j h_j^k - c \sum_i z_i$, where $\bullet$ denotes element-wise product followed by summation and $*$ denotes convolution. Each hidden group $\mathbf{h}_k$ is connected to the visible units in $\mathbf{z}$ with a convolutional filter $\mathbf{w}_k$. To lower the amount of parameters to be estimated, we let each element in $\mathbf{w}_k$ model two neighboring elements in $\mathbf{z}$, e.g. a filter of size 7 will span over 14 voxels in $\mathbf{z}$. Furthermore, each hidden group has a bias b_k and $\mathbf{z}$ a bias c.

We separately train one RBM for the complete tumor label $\mathbf{z}$ and one RBM for the tumor core label $\mathbf{y}$, where we estimate the filters and bias terms from training data. This is done by stochastic gradient ascent with contrastive divergence approximation of the log-likelihood gradients with one Gibbs sample step

[2]. We use the enhanced gradient to obtain more distinct filters [3]. After the training phase we combine the two RBMs to form the tumor shape prior:

$$p(\mathbf{z}, \mathbf{y}, \mathbf{H}, \mathbf{G}) \propto \mathrm{e}^{-E(\mathbf{z},\mathbf{H})-E(\mathbf{y},\mathbf{G})-f(\mathbf{y},\mathbf{z})}. \tag{3}$$

For each voxel, $f(y_i, z_i) = \infty$ if $y_i = 1$ and $z_i = 0$, and otherwise 0. This restricts tumor core tissue to only exist within the complete tumor.

Inference: We initially estimate $\boldsymbol{\theta}$ by a generalized Expectation-Maximization algorithm (GEM), where the tumor shape prior's energy is replaced with a simple energy of the form: $-\sum_i [l_i \neq BG](z_i \log w + (1-z_i)\log(1-w))$. This reduces the model to the same as in [4] with the addition of $p(\boldsymbol{\theta})$. We set w to the expected fraction of tumor tissue within brain tissue, estimated from training data. After the initial parameter estimation, we fix the bias field parameters and infer the remaining variables by block-Gibbs Markov chain Monte Carlo sampling (MCMC). This is straightforward to implement as each of the conditional distributions $p(\mathbf{l}, \mathbf{z}, \mathbf{y}|\mathbf{D}, \mathbf{H}, \mathbf{G}, \boldsymbol{\theta})$, $p(\mathbf{H}|\mathbf{z})$, $p(\mathbf{G}|\mathbf{y})$ and $p(\boldsymbol{\theta}|\mathbf{D}, \mathbf{l}, \mathbf{z}, \mathbf{y})$ factorizes over its components. The MCMC is initialized with a *maximum a posteriori* (MAP) segmentation after GEM. After a burn-in period, we collect samples of $\mathbf{l}, \mathbf{z}$ and $\mathbf{y}$ and perform a voxel-wise majority voting across the collected samples.

3 Experiments

We used the training data of the BRATS 2013 challenge (30 subjects) as our training data set and tested the proposed method on the two test sets of 2013 (Leaderboard: 25 subjects, Challenge: 10 subjects) [7] and the training data of the 2015 BRATS challenge (274 subjects, some are re-scans). The data include four MR-sequences: FLAIR, T1, T2 and contrast-enhanced T1, and ground truth segmentations. All data have previously been skull-stripped.

Implementation: We used 40 filters of size $(7 \times 7 \times 7)$ for each RBM, trained with 9600 gradient steps of size 0.1, which took around 3 days each. To extend the training data, the tumor segmentations were flipped in 8 directions.

We registered the healthy tissue atlas by an affine transformation and log-transformed the MR intensities, to account for the additive bias field model [4]. We represented the core label $\mathbf{y}$ with one Gaussian during GEM, corresponding to enhanced core, and two during MCMC, one for enhanced core and one for remaining core. Before MCMC, the remaining core Gaussian was initialized by randomly setting $y_i = 1$ to a fraction of the voxels with $z_i = 1$ and $y_i = 0$ in the MAP segmentation. The fraction was chosen so that the total fraction of core within the complete tumor equaled the average fraction in the training data set. All other labels were represented by one Gaussian each, except CSF and BG that were represented with two Gaussians each.

Due to the large size variation of tumors, we found it beneficial to alter the bias term c connected to $\mathbf{z}$ to better represent the tumor to be segmented. Before MCMC, we added $\log\left(\frac{p_{zs}(1-p_{zt})}{p_{zt}(1-p_{zs})}\right)$ to $\mathbf{c}$, where p_{zs} denotes the fraction of tumor within the GEM-segmented brain and p_{zt} denotes the average tumor size in the training data set, used to train the RBM. We altered the bias term connected to $\mathbf{y}$ in the same way, with the difference that we instead used the average fraction of core within complete tumor in the training data set.

The full segmentation algorithm took approximately 30 minutes per subject. We generated 15 samples after a burn-in of 200. All computations were done on a i7-5930K CPU and a GeForce GTX Titan Black GPU in MATLAB 2014b.

Results: At the time of writing, our method is ranked in the top-5 of all submitted results to the BRATS 2013 evaluation platform [8]. It performed well on complete tumor (rank 2 on both data sets) and core (rank 2 and 3), but not as well on enhanced core (rank 9). The lower performance on enhanced core is not surprising, as we base the segmentation on one Gaussian without any prior to separate it from the rest of the core. Average Dice scores and robust Hausdorff distances (95% quantile) on all data sets are shown in table 1. The results on the 2015 training data set are lower, as it includes more difficult subjects with substantial artifacts, more progressed tumors and resections.

	Dice [%]						Hausdorff [mm]		
Data sets	Comp.,	*HG/LG*	Core,	*HG/LG*	Enh.,	*HG/LG*	Comp.	Core	Enh.
2015 Training	77 ± 19	*76/78*	64 ± 29	*69/44*	52 ± 33	*58/31*	18	17	15
2013 Challenge	87 ± 3	*87/–*	82 ± 15	*82/–*	70 ± 15	*70/–*	–	–	–
2013 Leaderb.	83 ± 17	*87/59*	71 ± 27	*78/32*	54 ± 51	*64/0*	–	–	–

Table 1. Average Dice and Hausdorff scores. Hausdorff for enhanced core excludes 12 subjects due to missing label in either the ground truth or estimated segmentation.

Acknowledgements: This research was supported by NIH NCRR (P41-RR14075), NIBIB (R01EB013565) and the Lundbeck Foundation (R141-2013-13117).

References

1. Lee, H., Grosse, R., Ranganath, R., Ng, A. Y.: Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In: Proceedings of the 26th Annual International Conference on Machine Learning, ACM (2009)
2. Fischer, A., Igel, C.: Training restricted Boltzmann machines: An introduction. Pattern Recognition 47(1) (2014) 25-39
3. Melchior, J., Fischer, A., Wang, N., Wiskott, L.: How to Center Binary Restricted Boltzmann Machines. arXiv preprint arXiv:1311.1354 (2013)
4. Van Leemput, K., Maes, F., Vandermeulen, D., Suetens, P.: Automated model-based tissue classification of MR images of the brain. IEEE Transactions on Medical Imaging 18(10) (1999)
5. Ashburner, J., Friston, K., Holmes, A., Poline, J.-B.: Statistical Parametric Mapping. The Wellcome Dept. Cognitive Neurology, Univ. College London, London, U.K. Available: http://www.fil.ion.ucl.ac.uk/spm/
6. Murphy, K. P.: Machine learning: a probabilistic perspective. MIT Press (2012)
7. Menze, B. H., et al.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). To appear in IEEE Transactions on Medical Imaging (2015)
8. Kistler, M., Bonaretti, S., Pfahrer, M., Niklaus, R., Büchler, P.: The virtual skeleton database: an open access repository for biomedical research and collaboration. Journal of Medical Internet Research 15(11) (2013)

Segmentation of Gliomas in Multimodal Magnetic Resonance Imaging Volumes Based on a Hybrid Generative-Discriminative Framework

Spyridon Bakas, Ke Zeng, Aristeidis Sotiras, Saima Rathore, Hamed Akbari, Bilwaj Gaonkar, Martin Rozycki, Sarthak Pati, and Christos Davatzikos

Section of Biomedical Image Analysis, Center for Biomedical Image Computing and Analytics, Perelman School of Medicine, University of Pennsylvania, USA.

Abstract. We present an approach for segmenting low- and high-grade gliomas in multimodal magnetic resonance imaging volumes. The proposed approach is based on a hybrid generative-discriminative model. First, a generative approach based on an Expectation-Maximization framework that incorporates a glioma growth model is used to segment the scans into tumor, as well as healthy tissue labels. Then, a gradient boosting multi-class classification scheme is used to refine tumor labels. Lastly, a probabilistic Bayesian strategy is employed to finalize the tumor segmentation based on patient-specific intensity statistics from the multiple modalities. We evaluated our approach in 186 cases and report promising results that demonstrate the potential of our approach.

Keywords: Segmentation, Brain Tumor, Glioma, Multimodal MRI, BraTS challenge, Gradient Boosting, Expectation Maximization, Brain Tumor Growth Model, Probabilistic Model

1 Introduction

Gliomas comprise a group of primary central nervous system (CNS) tumors of neuroglial cells (*e.g.*, astrocytes and oligodendrocytes) that have different degrees of aggressiveness. They are mainly divided into low- and high-grade gliomas (LGGs and HGGs) according to their progression rate and histopathology. LGGs are less common than HGGs, constitute approximately 20% of CNS glial tumors, and almost all of them eventually progress to HGGs [9]. HGGs are rapidly progressing malignancies, divided based on their histopathologic features into anaplastic gliomas and glioblastomas (GBMs) [13].

Gliomas consist of various parts, each of which shows a different imaging phenotype in multimodal magnetic resonance imaging (MRI). Typically, the core of HGGs consists of enhancing, non-enhancing and necrotic parts, whereas the core of LGGs does not necessarily include an enhancing part. Another critical feature, for both understanding and treating gliomas, is the peritumoral edematous region. Edema occurs from infiltrating tumor cells, as well as a biological response to the angiogenic and vascular permeability factors released by the spatially adjacent tumor cells [1].

Quantification of the various parts of gliomas, in multimodal MRI, has an important role in treatment decisions, planning, as well as monitoring in longitudinal studies. The accurate segmentation of these regions is required to allow this quantification. However, tumor segmentation is extremely challenging due to the tumor regions being defined through intensity changes relative to the surrounding normal tissue, and such intensity information being disseminated across various modalities for each region. Additional factors that contribute to the difficulty of brain tumor segmentation task is the motion of the patient during the examination, as well as the magnetic field inhomogeneities. Hence, the manual annotation of such boundaries is time-consuming, prone to misinterpretation, human error and observer bias [2], with intra- and inter-rater variability up to 20% and 28%, respectively [10]. Computer-aided segmentation of brain tumor images would thus be a important advancement. Towards this end, we present a computer-aided segmentation method that aims to accurately segment such tumors and eventually allow for their quantification.

The remained of this paper is organized as follows: Sec. 2 details the provided data, while Sec. 3 presents the proposed segmentation strategy. The experimental validation setting is described in Sec. 4 along with the obtained results. Finally, Sec. 5 concludes the paper with a short discussion and potential future research directions.

2 Materials

The data used in this study describe 186 preoperative multimodal MRI scans of patients with gliomas (54 LGGs and 132 HGGs), provided as the training set for the multimodal Brain Tumor Segmentation (BraTS) 2015 challenge, from the Virtual Skeleton Database (VSD) [7]. Specifically, these data were a combination of the training set (10 LGGs and 20 HGGs) used in the BraTS 2013 challenge [11], as well as 44 LGG and 112 HGG scans provided from the National Institutes of Health (NIH) Cancer Imaging Archive (TCIA). The data of each patient consisted of native and contrast-enhanced (CE) T1-weighted, as well as T2-weighted and T2 Fluid-attenuated inversion recovery (FLAIR) MRI volumes. The volumes of the various modalities have been skull-stripped, co-registered to the same anatomical template and interpolated to $1mm^3$ voxel resolution.

To quantitatively evaluate the proposed method, ground truth (GT) segmentations for the training set were also provided. Specifically, the data from BraTS 2013 were manually annotated, whereas data from TCIA were automatically annotated by fusing the approved by experts results of the segmentation algorithms that ranked high in the BraTS 2012 and 2013 challenges [11]. The GT segmentations comprise the enhancing part of the tumor (ET), the tumor core (TC), which is described by the union of necrotic, non-enhancing and enhancing parts of the tumor, and the whole tumor (WT), which is the union of the TC and the peritumoral edematous region.

3 Methods

The provided image volumes are initially smoothed using a low-level image processing method, namely Smallest Univalue Segment Assimilating Nucleus (SUSAN) [12], to reduce intensity noise in regions of uniform intensity profile. Then, the intensity histograms of all volumes are matched to a reference volume.

A modified version of the GLioma Image SegmenTation and Registration (GLISTR) software [5] is subsequently used to delineate the boundaries of healthy and tumor tissues in the brain volume of each patient. More specifically, the following tissues are segmented: white matter, gray matter, cerebrospinal fluid, vessels, cerebellum, edema, necrosis, non-enhancing and enhancing tumor. This modified version is semi-automatic and requires as input a single seed point and a radius for each tumor, as well as multiple points for modeling the intensity distribution of each brain tissue type. Given the single seed point and the radius, the bulk volume of each tumor is approximated by a sphere. The parametric model of the sphere is then used to initiate a brain tumor growth model [6] in order to approximate the deformation occurred to the surrounding brain tissues, due to the effect of the tumor's mass. This is implemented under an Expectation-Maximization framework, optimized jointly with the segmentation of the surrounding brain tissues, as described in [8]. The method produces a probability map for each tissue type and a label map, which is a very good initial segmentation of all different tissues within a patient's brain.

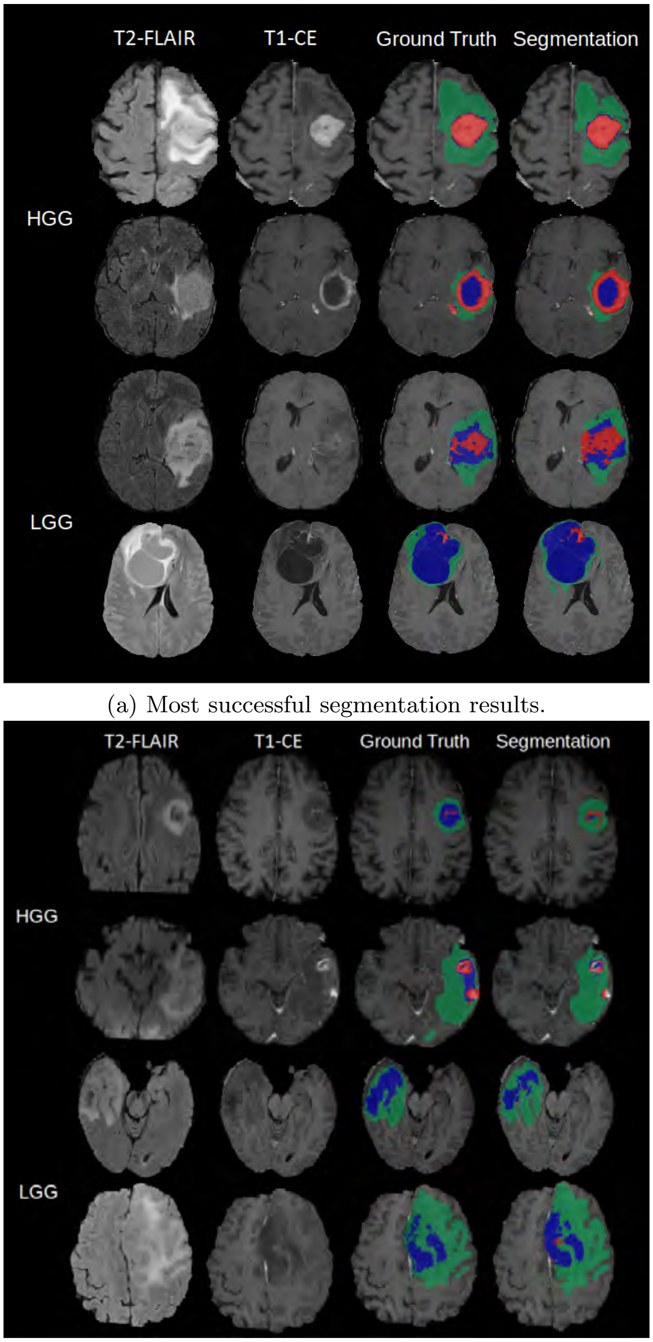
A machine-learning approach is then used to refine GLISTR results by utilizing information across multiple patients. Specifically, the gradient boosting algorithm [3] for voxel-level multi-label classification was employed, with deviance as the loss function. At each iteration, a decision tree of maximum depth 3 is added to the decision function, approximating the current negative gradient of the objective. Randomness is introduced when constructing each tree [4]. Each decision tree is fit to a subsample of the training set, with sampling rate set to 0.6, and the split is determined among a randomly selected number of features at each node with the number of chosen features proportional to the square root of the total number of features. The algorithm is terminated after 100 such iterations. Furthermore, the features used for training our model comprise the geodesic distance of each voxel (v_i) from the tumor seed point used by GLISTR, the intensity value of each voxel ($I(v_i)$) and their differences among all four modalities (*i.e.*, T1, T1-CE, T2, T2-FLAIR), Laplacian of Gaussian, image gradient magnitude, the GLISTR probability maps, and first and second order texture statistics computed from a graylevel co-occurrence matrix. It should also be mentioned that our model was trained using both LGG and HGG samples simultaneously using a 54-fold cross-validation setting (given that 54 LGGs are present in the data). The cross-validation setting is necessary in order to avoid over-fitting.

Finally, a patient-wise refinement is performed by assessing the local intensity distribution of the current segmentation labels and updating their spatial configuration based on a probabilistic model. Firstly, the intensity distribution of voxels with GLISTR posterior probability equal to 1 for the tissue classes of white matter, edema, necrosis, non-enhancing and enhancing tumor, are populated separately. Note that in the current segmentation goal, there is no distinction between the non-enhancing and the necrotic parts of the tumor. A normalization to the histograms of pair-wise distributions is then applied. The class-conditional probability densities ($Pr(I(v_i)|Class_1)$ and $Pr(I(v_i)|Class_2)$) are modeled by fitting distinct Gaussian models, using Maximum Likelihood Estimation to find the mean and standard deviation for each class. There are three pair-wise distributions considered here; the edema voxels opposed to the white matter voxels in the T2-FLAIR volume, the ET voxels opposed to the edema voxels in the T1-CE volume, and the ET voxels opposed to the union of the necrosis and the non-enhancing tumor in the T1-CE volume. In all cases, the former intensity population is expected to have much higher (*i.e.*, brighter) values. Hence, voxels of each class with small spatial proximity to the opposing tissue class are evaluated based on their intensity. Specifically, the intensity $I(v_i)$ of each of these voxels is assessed and $Pr(I(v_i)|Class_1)$ is compared with $Pr(I(v_i)|Class_2)$. This voxel, v_i, is then classified into a tissue class according to the larger of the two conditional probabilities. This is equivalent to a classification based on Bayes Theorem with equal priors for the two classes, *i.e.*, $Pr(Class_1) = Pr(Class_2) = 0.5$.

4 Experiments and Results

In order to assess the segmentation performance of our method, we evaluated the overlap between the proposed tumor labels and the GT in three regions, *i.e.*, WT, TC and ET, as suggested in [11]. Fig. 1 showcases example segmentation results along with the respective GT segmentations for eight patients (four HGGs and four LGGs). These correspond to the two most and least successful segmentation results for each glioma grade. we observe high agreement between the generated results and the provided labels. We note that the highest overlap is observed for edema, while there is some disagreement between the segmentations of the enhancing and non-enhancing parts of the tumor.

To further appraise the performance of the proposed method, we quantitatively validated the per-voxel overlap between respective regions using the DICE coefficient (see Fig. 2 and Table 1). This metric takes values between 0 and 1, with higher values corresponding to increased overlap. Moreover, aiming to understand fully the obtained results, we stratified them based on the labeling protocol of the GT segmentation. In particular, data with manually annotated GT (*i.e.*, BraTS 2013 data) was evaluated separately from data with automatically defined GT (*i.e.*, TCIA data). The reason behind this distinction is twofold. First, only manual segmentation can be considered as gold standard, thus allowing us to evaluate the potential of our approach when targeting an interactive clinical work-flow. Second, results validated using automatically defined GT should be interpreted with caution because of the inherently introduced bias towards the employed automated methods, which also influences visually inspecting experts [2]. As a consequence, our method may be negatively im-



(a) Most successful segmentation results.

(b) Least successful segmentation results.

Fig. 1. Examples for four LGG and four HGG patients. Green, red and blue masks denote the edema, the enhancing tumor and the union of the necrotic and non-enhancing parts of the tumor, respectively.

pacted since it may learn to reproduce the systematic mistakes of the provided annotations.

Fig. 2 reports the distributions of the DICE score across patients for each step of the proposed method and for each tissue label (WT, TC and ET) while Table 1 reports the respective mean and median values. The results are presented following the previously described stratification. Fig. 2 shows a clear step-wise improvement in both the mean and median values of all tissue labels when considering either the complete set of data or the automatically segmented one. On the contrary, we observe a step-wise deterioration of both the mean and median values for the TC label when assessing the manually annotated subset of the data (see Table 1 for the exact values). This is probably the effect of learning systematically mislabeled voxels present in the automatically generated GT segmentations (see mislabeled ET in GT of the second HGG in Fig. 1.(a)).

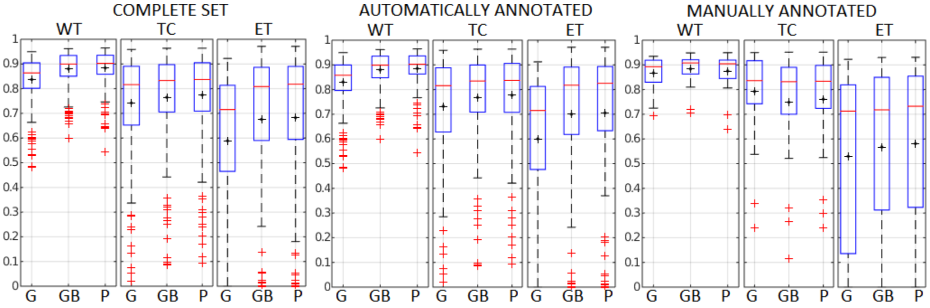


Fig. 2. Distributions of the DICE score across patients for each step of the proposed method, each tissue label and different groupings of data. The black cross and the red line inside each box denote the mean and median values, respectively.

5 Discussion

We presented an approach that combines generative and discriminative methods towards providing a reliable and highly accurate segmentation of LGGs and HGGs in multimodal MRI volumes. Our proposed approach is built upon the brain segmentation results provided by a modified version of GLISTR. GLISTR segments the brain into tumor and healthy tissue labels by means of a generative model encompassing a tumor growth model and a probabilistic atlas of healthy individuals. GLISTR tumor labels are subsequently refined taking into account population-wide tumor label appearance statistics that were learned by employing a gradient boosting multi-class classifier. The final results were produced by adapting the segmentation labels based on patient-specific label intensity distributions from the multiple modalities.

Table 1. Mean and median values of the DICE score for each step of the proposed method, each tissue label and different groupings of data.

Data	Method	Dice score (mean)			Dice score (median)		
		WT	TC	ET	WT	TC	ET
complete training set (n=186)	**GLISTR**	83.7%	74.2%	58.6%	86.4%	81.6%	71.6%
	GLISTR+GB	87.9%	76.5%	67.6%	89.9%	83.3%	80.9%
	Proposed	**88.4%**	**77.4%**	**68.2%**	**90.3%**	**83.7%**	**82%**
manually annotated (n=30)	**GLISTR**	86.7%	79.2%	52.9%	89.2%	83.6%	71.26%
	GLISTR+GB	**88.3%**	74.8%	56.7%	**90.8%**	83.2%	72.6%
	Proposed	87.6%	**76.1%**	**58.1%**	90.5%	**83.4%**	**75.7%**
automatically annotated (n=156)	**GLISTR**	83.1%	73.2%	60.1%	85.8%	81.6%	71.6%
	GLISTR+GB	87.9%	76.8%	70.5%	89.9%	83.5%	82.6%
	Proposed	**88.5%**	**77.7%**	**71%**	**90.3%**	**83.7%**	**82.8%**

Our approach was able to deliver high quality tumor segmentation results by significantly improving GLISTR results through the adopted post-processing strategies. This improvement was evident for both manually and automatically segmented data. The only case where the post-processing resulted in a decrease of the performance is for the TC label when considering only the manually segmented data. This could be probably attributed to the fact that the supervised gradient boosting model learned consistent errors present in the automatically generated segmentations and propagated them when refining GLISTR results. While pooling information for more patients seems to be benefiting the learning algorithm, it also introduces a bias towards the more numerous automatically generated data. Accounting for this bias by weighting accordingly manually and automatically segmented samples could possible allow for harnessing the additional information without compromising quality.

The proposed approach segmented the whole tumor and the tumor core with high accuracy for both LGGs and HGGs. However, the segmentation of the enhancing tumor could be further improved considering that gliomas can be distinguished into two distinct imaging phenotypes, which are not necessarily consistent with their clinical grade (*i.e.*, LGG/HGG). This is due to the fact that LGGs are characterized by a distinct pathophysiological phenotype that is often marked by the lack of an enhancing part, hence not having the same imaging phenotype with the HGGs. These imaging signatures could be possibly exploited in a machine learning framework that considers separately radiologically defined HGGs and LGGs, *i.e.*, tumors with and without a distinctive enhancing part. By modeling separately these distinct imaging phenotypes, the goal will be to capture better the imaging heterogeneity and improve label prediction in the BraTS 2015 testing set.

References

[1] Akbari, H., Macyszyn, L., Da, X., Wolf, R.L., Bilello, M., Verma, R., et.al.: Pattern analysis of dynamic susceptibility contrast-enhanced MR imaging demonstrates peritumoral tissue heterogeneity. Radiology 273(2), 502–510 (2014)

[2] Deeley, M.A., Chen, A., Datteri, R., Noble, J.H., Cmelak, A.J., Donnelly, E.F., et.al.: Comparison of manual and automatic segmentation methods for brain structures in the presence of space-occupying lesions: a multi-expert study. Phys Med Biol 56(14), 4557–4577 (2011)

[3] Friedman, J.H.: Greedy function approximation: A gradient boosting machine. Ann Statist 29(5), 1189–1232 (2001)

[4] Friedman, J.H.: Stochastic gradient boosting. Computational Statistics & Data Analysis 38(4), 367–378 (2002)

[5] Gooya, A., Pohl, K.M., Bilello, M., Cirillo, L., Biros, G., Melhem, E.R., et.al.: GLISTR: Glioma Image Segmentation and Registration. IEEE Trans Med Imaging 31(10), 1941–1954 (2012)

[6] Hogea, C., Davatzikos, C., Biros, G.: An image-driven parameter estimation problem for a reactiondiffusion glioma growth model with mass effects. J Math Biol 56(6), 793–825 (2008)

[7] Kistler, M., Bonaretti, S., Pfahrer, M., Niklaus, R., Büchler, P.: The Virtual Skeleton Database: An Open Access Repository for Biomedical Research and Collaboration. J Med Internet Res 15(11), e245 (2013)

[8] Kwon, D., Shinohara, R.T., Akbari, H., Davatzikos, C.: Combining Generative Models for Multifocal Glioma Segmentation and Registration. Medical Image Computing and Computer-Assisted Interventions 17(1), 763–770 (2014)

[9] Louis, D.N.: Molecular pathology of malignant gliomas. Annu Rev Pathol 1, 97–117 (2006)

[10] Mazzara, G.P., Velthuizen, R.P., Pearlman, J.L., Greenberg, H.M., Wagner, H.: Brain tumor target volume determination for radiation treatment planning through automated MRI segmentations. Int J Radiat Oncol Biol Phys 59(1), 300–312 (2004)

[11] Menze, B., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., et. al.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Trans Med Imaging p. 33 (2014)

[12] Smith, S.M., Brady, J.M.: SUSAN - a new approach to low level image processing. Int Journal of Computer Vision 23(1), 45–78 (1997)

[13] Wen, P.Y., Kesari, S.: Malignant gliomas in adults. N Engl J Med 359(5), 492–507 (2008)

Structured Prediction with Convolutional Neural Networks for Multimodal Brain Tumor Segmentation

Pavel Dvořák[1,2] and Bjoern Menze[3]

[1] Dept. of Telecommunications,
Faculty of Electrical Engineering and Communication,
Brno University of Technology, Czech Republic;
[2] ASCR, Institute of Scientific Instruments,
Královopolská 147, 612 64 Brno, Czech Republic
[3] Institute for Advanced Study and Department of Computer Science,
TU München, Germany
pavel.dvorak@phd.feec.vutbr.cz,bjoern.menze@tum.de

Abstract. Most medical images feature a high similarity in the intensities of nearby pixels and a strong correlation of intensity profiles across different image modalities. One way of dealing with – and even exploiting – this correlation is the use of local image patches. In the same way, there is a high correlation between nearby labels in image annotation, a feature that has been used in the "local structure prediction" of local label patches. In the present study we test this local structure prediction approach for 3D segmentation tasks, systematically evaluating different parameters that are relevant for the dense annotation of anatomical structures. We choose convolutional neural network as learning algorithm, as it is known to be suited for dealing with correlation between features. We evaluate our approach on the public BRATS2014 data set with three multimodal segmentation tasks, being able to obtain state-of-the-art results for this brain tumor segmentation data set consisting of 254 multimodal volumes with computing time of only 13 seconds per volume.

Keywords: Brain Tumor, Clustering, CNN, Deep Learning, Image Segmentation, MRI, Patch, Structure, Structured Prediction.

1 Introduction

Medical images show a high correlation between the intensities of nearby voxels and the intensity patterns of different image modalities acquired from the same volume. Patch-based prediction approaches make use of this local correlation and rely on dictionaries with finite sets of image patches. They succeed in a wide range of application such as image denoising, reconstruction, and even the synthesis of image modalities for given applications [6]. Moreover, they were used successfully for image segmentation, predicting the most likely label of the voxel in the center of a patch [17]. All of these approaches exploit the redundancy of local image information and similarity of *image features* in nearby pixels or voxels. For most applications, however, the same local similarity is present among the *image labels*, e.g., indicating the extension of underlying anatomical structure. This structure has already been used in medical imaging but only at *global level*, where the shape of the whole segmented structure is considered, e.g. [13] or [21]. Here we will focus on *local structure* since global structure is not applicable for objects with various shape and location such as brain tumors.

Different approaches have been brought forward that all make use of the local structure of voxel-wise image labels. Zhu et al. [22] proposed a recursive segmentation approach with recognition templates in multiple layers to predict extended 2D patches instead of pixel-wise labels. Kontschieder et al. [8] extended the previous work with structured image labeling using random forest. They introduced a novel data splitting function, based on random pixel position in a patch, and exploited the joint distributions of structured labels. Chen et al. [2] introduced techniques for image representation using a shape epitome dictionary created by affinity propagation, and applied it together with a conditional random field models for image labeling. Dollar et al. [4] used this idea in edge detection using k-means clustering in label space to generate an edge dictionary, and a random forest classification to predict the most likely local edge shape.

In spite of the success of patch-based labeling in medical image annotation, and the highly repetitive local label structure in many applications, the concept of patch-based local structure prediction, i.e., the prediction of extended label patches, has not received attention in the processing of 3D medical image yet. However, approaches labeling supervoxels rather than voxels has already appeared, e.g. hierarchical segmentation by weighted aggregation extended into 3D by Akselrod-Ballin et al. [1] and later by Corso et al. [3], or spatially adaptive random forests introduced by Geremia et al. [5].

In this paper, we will transfer the idea of *local structure prediction* [4] using patch-based label dictionaries to the task of dense labels of pathological structures in multimodal 3D volumes. Different from Dollar, we will use convolutional neural networks (CNNs) for predicting label patches as CNNs are well suited for dealing with local correlation, also in 3D medical image annotation tasks [9, 14]. We will evaluate the local structure prediction of label patches on a public data set with several multimodal segmentation subtasks, i.e., on the 2014 data set of the Brain Tumor Image Segmentation Challenge [11], where a CNN outperformed other approaches [19]. In this paper, we focus on evaluating design choices for local structure prediction and optimize them for reference image segmentation task in medical image computing.

Brain tumor segmentation is a challenging task that has attracted some attention over the past years. It consists of identifying different tumor regions in a set of multimodal tumor images: the whole tumor, the tumor core, and the active tumor [11]. Algorithms developed for brain tumor segmentation task can be classified into two categories: Generative models use a prior knowledge about the spatial distribution of tissues and their appearance, e.g. [15, 7], which requires accurate registration with probabilistic atlas encoding prior knowledge about spatial structure at the organ scale [10]. Our method belongs to the group of *discriminative models*. Such algorithms learn all the characteristics from manually annotated data. In order to be robust, they require substantial amount of training data [20, 23].

In the following, we will describe our local structure prediction approach (Sec. 2), and present its application to multimodal brain tumor segmentation (Sec. 3). Here we will identify, analyze, and optimize the relevant model parameters of the local structure prediction for all different sub-tasks and test the final model on clinical test set, before offering conclusion (Sec. 4).

2 Methods

The brain tumor segmentation problem consists of three sub-problems: identifying the whole tumor region in a set of multimodal images, the tumor core region, and the active tumor region [11]. All three sub-tasks are process separately, which changes the multi-class segmentation task into three binary segmentation sub-tasks.

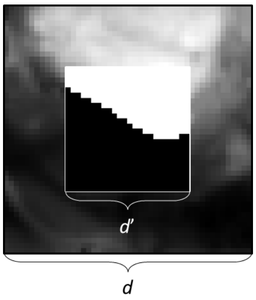


Fig. 1. Local structure prediction: Image feature patches (with side length d) are used to predict the most likely label patch (with side length d') in its center. While standard patch based prediction approaches use $d' = 1$ (voxel), we consider in this paper all values with $1 \leq d' \leq d$.

Structured prediction. Let $\mathbf{x}$ be the *image patch* of size $d \times d$ from image space $\mathcal{I}$. Focusing on 2D patches, a patch $\mathbf{x}$ is represented as $\mathbf{x}(u, v, I)$ where (u, v) denotes the patch top left corner coordinates in multimodal image $I(s, V)$ where s denotes the slice position in multimodal volume V.

Label patches. Treating the annotation task for each class individually, we obtain a label space $\mathcal{L} = \{0, 1\}$ that is given by an expert's manual segmentation of the pathological structures. The *label patch* is then a patch $\mathbf{p}$ of size $d' \times d'$ from the structured label space $\mathcal{P}$, i.e. $\mathcal{P} = \mathcal{L}^{d' \times d'}$. The label size d' is equal or smaller than the image patch size d. The label patch $\mathbf{p}$ is centered on its corresponding image patch $\mathbf{x}$ (Fig. 1), and it is represented as $\mathbf{p}(u + m, v + m, L)$ where $L(s, W)$ is a manual segmentation in slice s of label volume W and m denotes the margin defined as $m = \frac{1}{2}(d - d')$.

Optimal values for d and d' and, hence, the ratio $r = \frac{d'}{d}$ may vary depending on the structure to be segmented and the image resolution.

Generating the label patch dictionary. We cluster label patches $\mathbf{p}$ into N groups using k-means leading to a label patch dictionary of size N. Subsequently, the *label template* $\mathbf{t}$ of group n is identified as the average label patch of given cluster. In the segmentation process, these smooth label templates $\mathbf{t}$ are then used for the segmentation map computation rather than strict border prediction as used in previous local structure prediction methods [2, 8, 22]. The structures are learned directly from the training data instead of using predefined groups as in [22]. Examples of ground truth label patches with their representation by a dictionary of size $N = 2$ (corresponding to common segmentation approach) and $N = 32$ is depicted in Fig. 2.

The size of label patch dictionary N and, hence, the number of classes in the classification problem, may differ between problems depending on variability and shape complexity of the data.

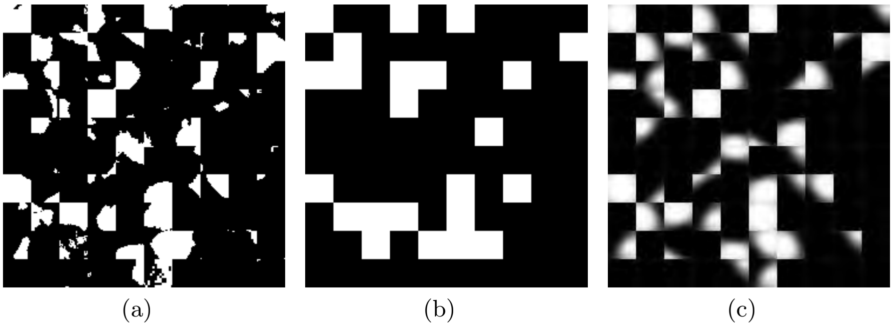


Fig. 2. Ground truth label patches (a) with corresponding binary representation indicating label at the central pixel (b), and structured (c) representation.

Defining the N-class prediction problem. After we have obtained a set of N clusters, we transform our binary segmentation problem into an N class prediction task: We identify each training image patch $\mathbf{x}$ with the group n that the corresponding label patch $\mathbf{p}$ has been assigned to during the label patch dictionary generation. In prediction, the label template $\mathbf{t}$ of the predicted group n (size $d' \times d'$) is assigned to the location of each image patch and all overlapping predictions of a neighborhood are averaged. According to the experiments a discrete threshold $th = 0.5$ was chosen for the final label prediction.

Convolutional Neural Network. We choose CNN as it has the advantage of preserving the spatial structure of the input, e.g., 2D grid for images. CNN consists of convolutional and pooling layers, usually applied in an alternating order. The CNN architecture used in this work is depicted in Fig. 3. It consists of two convolutional and two mean-pooling layers in alternating order. In both convolutional layers, we use 24 convolutional filters of kernel size 5×5. The input of the network is an image patch of size $4 \times d \times d$ (four MR modalities are present in multimodal volumes) and the output is a vector of length N indicating membership to one of the N classes in the label patch dictionary.

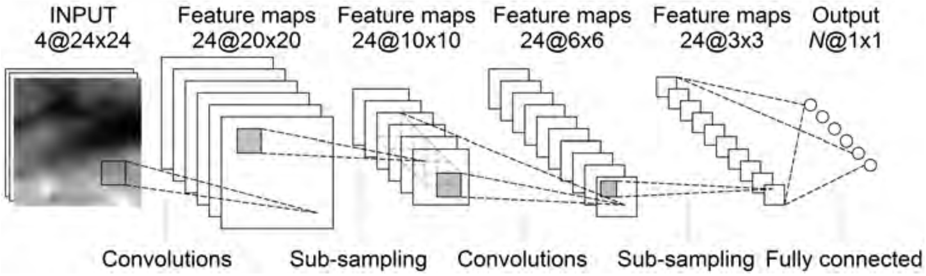


Fig. 3. Architecture of Convolutional Neural Network for $d = 24$. The input of the network is a multimodal image patch. The output of the network are N probabilities, where N denotes the size of label patch dictionary.

Slice Inference. Image patches from each multimodal volume are mapped into four 2D input channels of the network, similar to RGB image mapping. During the training phase, patches of given size are extracted from training volumes. Using the same approach for testing is inefficient and therefore different approach used in [12] is employed instead. The whole input 2D slice is fed to the network architecture, which leads to much faster convolution process than applying the same convolution several times to small patches. This requires proper slice padding by to be able to label pixels close to slice border.

The output of the network is a map of label scores. However, this label map is smaller than the input slice due to pooling layers inside the CNN architecture. In our case with two 2×2 pooling layers, there is only one value for every 4×4 region. Pinheiro and Collobert [12] fed the network by several versions of input image shifted on X and Y axis and merged the outputs properly. More common approach is to upscale the label map to the size of the input image. The latter approach is faster due to only one convolution per slice compared to 16 using the former approach in our case. Both of them were tested and will be compared.

One can see the sequential processing of the input multimodal slice in Fig. 4. 4(b) and 4(c) depict 24 outputs of the first and the second convolutional layers of CNN. 4(d) shows the final classification map of the CNN architecture. Note the average labels for given group in 4(e). One can compare them to the ground truth tumor border in the input image. The final probability map of the whole tumor area is depicted in 4(f).

Since the hierarchy exist between particular segmentation sub-tasks, both tumor core and active tumor are segmented only inside the whole tumor region. This makes the segmentation process much faster. Although the hierarchy exist between tumor core and active tumor as well, this approach is not used here since the segmentation of tumor core is the most difficult sub-task and usually the least accurate one.

Feature Representation. Before the processing of the data, the N4 bias field correction [18] is applied and the image intensities of brain are normalized

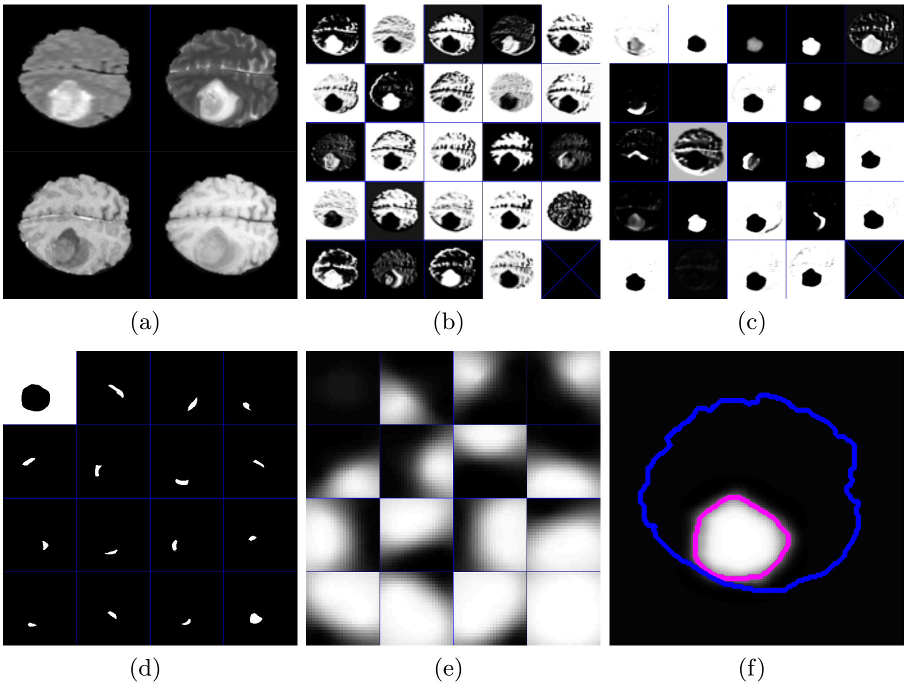


Fig. 4. Sequential processing of multimodal slice (a). (b) and (c) show all 24 outputs of the first and the second convolutional layer. (d) depicts the output of the whole CNN architecture for given 16 groups with average patch labels depicted in (e). (f) shows the final probability map of the whole tumor area with outlined brain mask (blue) and final segmentation (magenta) obtained by thresholding at 50% probability.

by their average intensity and standard deviation. All volumes in the BRATS database have the same dimension order and isotropic resolution, therefore the axial slice extraction is straightforward and no pre-processing step to get images in a given orientation and spatial resolution is necessary.

As it has been shown in [14], the computational demands of 3D CNN are still out of scope for today's computers. Therefore, we focus on processing the volume sequentially in 2D in the plane with the highest resolution, in our case the axial plane. Image patches from each multimodal volume are mapped into four 2D input channels of the network. This approach gives a good opportunity for parallelization of this task to reduce the run-time. Alternatives to this basic approach have been proposed: Slice-wise 3D segmentation using CNN was used in [14, 16]. The former showed non-feasibility of using 3D CNN for larger cubic patches and proposed using of 2D CNN for each orthogonal plane separately. The later proposed extraction of corresponding patches for given pixel from each orthogonal plane and mapping them as separated feature maps. In our work, we have tested both of these approaches and compared them to the single slice approach that we chose.

3 Experiments

Brain tumor segmentation is a challenging task that has attracted some attention over the past years. We use the BRATS data set that consists of multiple segmentation sub-problems: identifying the whole tumor region in a set of multimodal images, the tumor core region, and the active tumor region [11].

Image Data. Brain tumor image data used in this work were obtained from the MICCAI 2014 Challenge on Multimodal Brain Tumor Image Segmentation (BRATS) training set.[4] The data contains real volumes of 252 high-grade and 57 low-grade glioma subjects. For each patient, co-registered T1, T2, FLAIR, and post-Gadolinium T1 MR volumes are available. These 309 subjects contain more measurement for some patients and only one measurement per patient was used by us. The data set was divided into three groups: training, validation and testing. Our training set consists of 130 high grade and 33 low grade glioma subjects, the validation set consists of 18 high grade and 7 low grade glioma subjects, and the testing set consists of 51 high grade and 15 low grade glioma subjects, summing up to 254 multimodal volumes of average size 240×240×155. From each training volume, 1500 random image patches with corresponding label patches were extracted summing up to 244 500 training image patches. The patches are extracted from the whole volume within the brain area with higher probability around the tumor area.

Parameter Optimization Beside the parameters of the convolutional architecture, there are parameters of our model: image patch size d, label patch size d', and size of label patch dictionary N. These parameters were tested with pre-optimized fixed network architecture depicted in Fig. 3, which consists of two convolutional layers, both with 24 convolutional filters of kernel size 5×5, and two mean-pooling layers in alternating order. The values selected for subsequent experiments are highlighted in graphs with red vertical line.

Image patch size. The image patch size d is an important parameter since the segmented structures have different sizes and therefore less or more information is necessary for label structure prediction. Figure 5 shows the Dice score for different patch sizes with their best label patch size. According to the graphs, $d = 8$ was selected for active part segmentation and $d = 24$ for tumor core and whole tumor. All three tests were performed for $N = 32$, which according to the previous tests is sufficiently enough for all patch sizes. The best results were in all cases achieved for $d' \geq \frac{1}{2}d$. The values selected for subsequent experiments are indicated by red vertical line.

Size of label patch dictionary. The size of label patch dictionary N influence differences between each label template $\mathbf{t}$ as well as the differences between

[4] http://www.braintumorsegmentation.org/

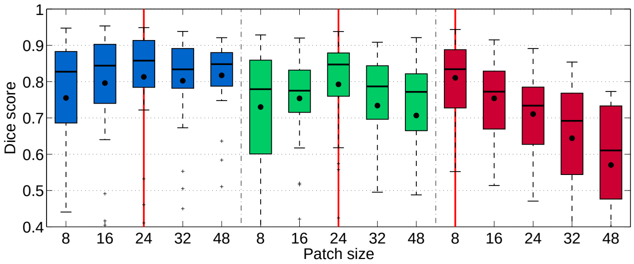


Fig. 5. Dice score as a function of the **image patch size** d with its best label patch size d' with label patch dictionary size $N = 32$ for the whole tumor (blue), tumor core (green) and active tumor(red).

belonging image patches $\mathbf{x}$ in each groups n. The results for several values of N are depicted in Fig. 6. Generally the best results were achieved for $N = 16$. The results were evaluated in similar manner as in the previous test, i.e. the best d' is used for each value of N. The values selected for subsequent experiments are indicated by red vertical line.

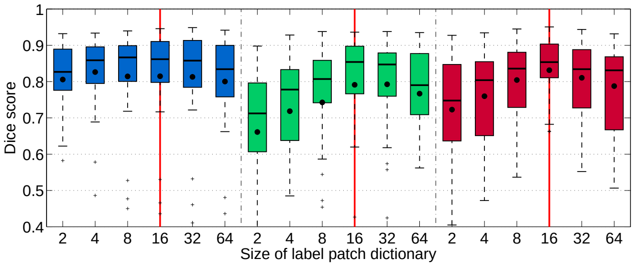


Fig. 6. Dice score as a function of **label patch dictionary size** N using the optima of Fig. 5: $d = 24$ for whole tumor (blue), $d = 24$ for tumor core (green), $d = 8$ for active tumor (red).

Label patch size. The label patch size d' influences the size of local structure prediction as well as the number of predictions for each voxel. Figure 7 shows the increasing performance with increasing d'. The values selected for subsequent experiments are indicated by red vertical line.

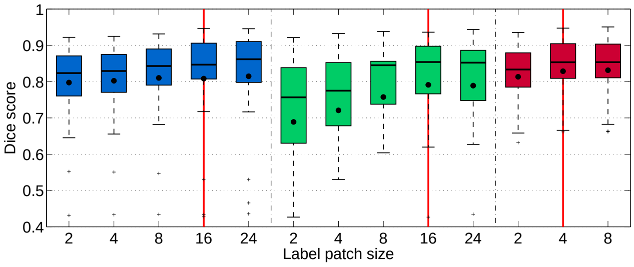


Fig. 7. Dice score as a function of **label patch size** d' for whole tumor (blue) with $d = 24$, tumor core (green) with $d = 24$, and active tumor (red) with $d = 8$, with label patch dictionary size $N = 16$.

2D versus 3D. We have tested both triplanar and 2.5D deep learning approaches for 3D data segmentation as proposed in [14] and [16], respectively, and compared them to single slice-wise segmentation. For both approaches, we have obtained about the same performance as for single slice-wise approach: the triplanar 2.5D segmentation decreased the performance by 2%, the 3D segmentation to a decrease of 5%. This observation is probably caused by lower resolution in sagittal and coronal planes.

Application to the test set. After the optimization of the parameters using validation set, we tested the algorithm on a new set of 66 subjects randomly chosen from BRATS 2014. The performance for both validation and test set of all three segmented structures is summarized in Tab. 1. For the test set, we achieved average Dice scores 83% (whole tumor), 75% (tumor core), and 77% (active tumor). The resulting Dice scores are comparable to intra-rater similarity that had been reported for the three annotation tasks in the BRATS data set [11] with Dice scores 85% (whole tumor), 75% (tumor core) and 74% (active tumor) and to the best results of automated segmentation algorithms with the Dice score of the top three in between 79%–82% (here: 83%) for the whole tumor segmentation task, 65%–70% (here: 75%) for the segmentation of the tumor core, and 58%–61% (here: 77%) for the segmentation of the active tumor region.

We show segmentations generated by our method and the ground truth segmentations for the three regions to be segmented on representative test cases in Fig. 8.

Table 1. Segmentation results on validation and test data sets, reporting average and median Dice scores. Shown are the results for all three segmented structures, i.e., whole tumor, tumor core and active tumor. Scores for active tumor are calculated for high grade cases only. "std" and "mad" denote standard deviation and median absolute deviance. HG and LG stand for high and low grade gliomas, respectively.

Dice Score (in %)	Whole	HG / LG	Core	HG / LG	Active
Validation set					
mean ± std	81±15	80±17 / 85±06	79±13	85±08 / 65±15	81±11
median ± mad	86±06	86±07 / 85±05	85±06	85±03 / 73±10	83±08
Test set					
mean ± std	83±13	86±09 / 76±21	75±20	79±14 / 61±29	77±18
median ± mad	88±04	88±03 / 87±05	83±08	82±07 / 72±14	83±09

Compute time vs accuracy. We have also tested the possibility of subsampling the volume in order to reduce the computational demands. The trade off between accuracy and computing time per volume is analyzed in Tab. 2 by running several experiments with different resolutions of the CNN output before final prediction of local structure (first column) as well as different distances between segmented slices (second column), i.e., different sizes of subsequent segmentation interpolation. All experiments were run on 4-core CPU Intel Xeon E3 3.30GHz. As one can see in the table, the state-of-the-art results can be achieved in an order of magnitude shorter time than in case of most methods participated in BRATS challenge. Thanks to fast implementation of the CNN segmentation, all three structures can be segmented in whole volume in 13 seconds without using GPU implementation. Processing by the CNN is approximately 80% of the overall computing time, while assigning final labels using local structure prediction requires only 17%. The rest of the time are other operations including interpolation. The overall training time, including label patch dictionary generation and training of all three networks using 20 training epochs, was approximately 21 hours.

Table 2. Tradeoff between spatial subsampling, computing time, and segmentation accuracy. First two columns express different CNN output resolution, i.e., after subsampling in x and y, and steps between segmented slices, i.e., after subsampling in z direction.

CNN output	Slice	Computing time	Dice Score (in%)		
resolution	step	per volume	Whole	Core	Active
1/4	4	13s	83	75	73
1/4	2	22s	84	75	74
1/4	1	74s	84	75	75
1/2	4	24s	83	75	74
1/2	2	41s	83	75	76
1/2	1	142s	84	75	76
1/1	4	47s	83	75	75
1/1	2	80s	83	75	77
1/1	1	280s	83	75	77

4 Conclusion

We have shown that exploiting local structure through the use of the label patch dictionaries improves segmentation performance over the standard approach predicting voxel wise labels. We showed that local structure prediction can be combined with, and improves upon, standard prediction methods, such as a CNN. When the label patch size optimized for a given segmentation task, it is capable of accumulating local evidence for a given label, and also performs a spatial regularization at the local level. On our reference benchmark set, our approach achieved state-of-the-art performance even without post-processing through Markov random fields which were part of most best performing approaches in the tumor segmentation challenge. Moreover, the all three structures can be extracted from the whole volume within only 13 seconds using CPU obtaining state-of-the-art results providing means, for example, to do online updates when aiming at an interactive segmentation.

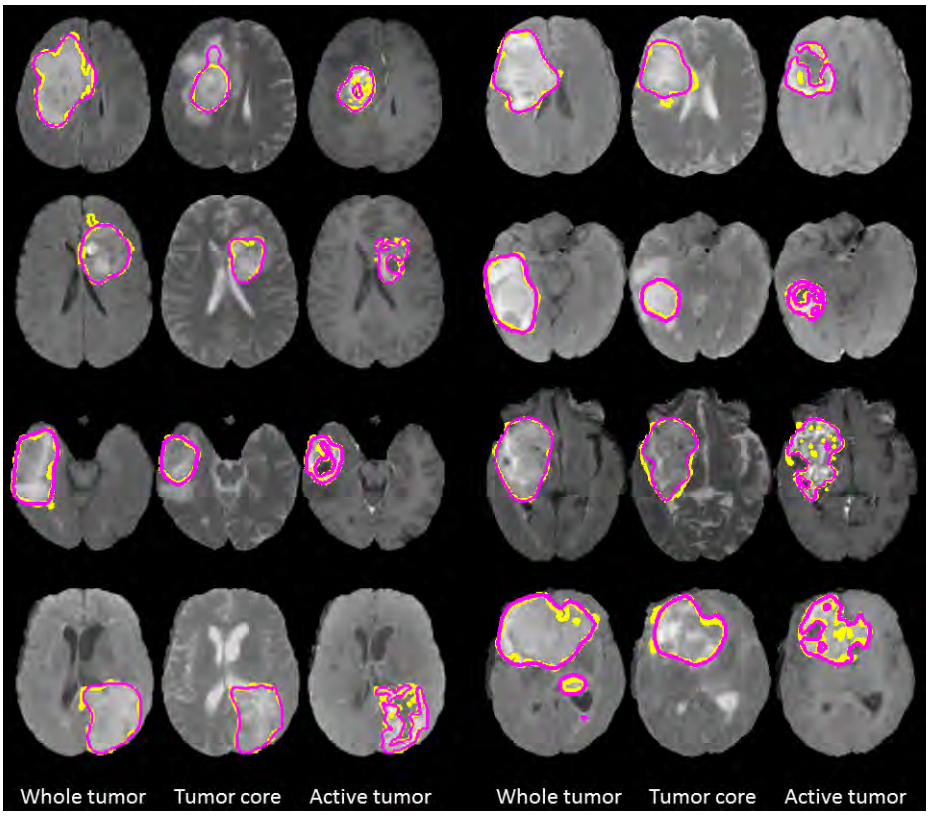


Fig. 8. Example of consensus expert annotation (yellow) and automatic segmentation (magenta) applied to the test image data set. Each row shows two cases. From left to right: segmentation of whole tumor (shown in FLAIR), tumor core (shown in T2) and active tumor (shown in T1c).

Acknowledgments. This work was partially supported through grants LO1401 and LD14091.

References

1. Akselrod-Ballin, A., et al.: An integrated segmentation and classification approach applied to multiple sclerosis analysis. In: Proc CVPR (2006)
2. Chen, L.C., Papandreou, G., Yuille, A.: Learning a dictionary of shape epitomes with applications to image labeling. In: Proc ICCV 2013. pp. 337–344 (2013)

3. Corso, J.J., et al.: Efficient multilevel brain tumor segmentation with integrated bayesian model classification. TMI 27(5), 629 – 640 (2011)
4. Dollar, P., Zittnick, C.L.: Structured forests for fast edge detection. In: Proc ICCV 2013. pp. 1841–1848 (2013)
5. Geremia, E., Menze, B.H., Ayache, N.: Spatially adaptive random forests. In: Proc ISBI (2013)
6. Iglesias, J.E., et al.: Is synthesizing MRI contrast useful for inter-modality analysis? In: Proc MICCAI 2013. pp. 631–638 (2013)
7. Kaus, M.R., et al.: Automated segmentation of mr images of brain tumors. Radiology 2018(2), 586–591 (2001)
8. Kontschieder, P., et al.: Structured class-labels in random forests for semantic image labelling. In: Proc ICCV 2011. pp. 2190–2197 (2011)
9. Liao, S., et al.: Representation learning: A unified deep learning framework for automatic prostate mr segmentation. In: Proc MICCAI 2013. pp. 254–261 (2013)
10. Menze, B., van Leemput, K., Lashkari, D., Weber, M.A., Ayache, N., Golland, P.: A generative model for brain tumor segmentation in multi-modal images. In: Proc MICCAI 2010, pp. 151–159 (2010), http://dx.doi.org/10.1007/978-3-642-15745-5_19
11. Menze, B., et al.: The multimodal brain tumor image segmentation benchmark (BRATS). IEEE TMI p. 33 (2014)
12. Pinheiro, P.H.O., Collobert, R.: Recurrent convolutional neural networks for scene labeling. In: International Conference on Machine Learning (ICML) (2014)
13. Pohl, K.M., et al.: A hierarchical algorithm for mr brain image parcellation. TMI 26(9), 1201–1212 (2007)
14. Prasoon, A., et al.: Deep feature learning for knee cartilage segmentation using a triplanar convolutional neural network. In: Proc MICCAI 2013. pp. 246–253 (2013)
15. Prastawa, M., Bullitt, E., Ho, S., Gerig, G.: A brain tumor segmentation framework based on outlier detection. Med Image Anal 8, 275–283 (2004)
16. Roth, H.R., Lu, L., Seff, A., et al.: A new 2.5D representation for lymph node detection using random sets of deep convolutional neural network observations. In: MICCAI. pp. 520–527 (2014)
17. Tong, T., et al.: Segmentation of MR images via discriminative dictionary learning and sparse coding: Application to hippocampus labeling. NeuroImage 76, 11–23 (2013)
18. Tustison, N., Avants, B., Cook, P., Gee, J.: N4itk: Improved n3 bias correction with robust b-spline approximation. In: Proc. of ISBI (2010)
19. Urban, G., et al.: Multi-modal brain tumor segmentation using deep convolutional neural networks. In: Proc MICCAI-BRATS. pp. 31–35 (2014)
20. Wels, M., Carneiro, G., Aplas, A., Huber, M., Hornegger, J., Co-maniciu, D.: A discriminative model-constrained graph cuts approach to fully automated pediatric brain tumor segmentation in 3d mri. In: Proc MICCAI. pp. 67–75 (2008)
21. Zhang, Y., Brady, M., Smith, S.: Segmentation of brain mr images through a hidden markov random field model and the expectation-maximization algorithm. TMI 20(1), 45–57 (2001)
22. Zhu, L., Chen, Y., Lin, Y., Lin, C., Yuille, A.L.: Recursive segmentation and recognition templates for 2d parsing. In: Koller, D., Schuurmans, D., Bengio, Y., Bottou, L. (eds.) NIPS, pp. 1985–1992 (2009)
23. Zikic, D., et al.: Decision forests for tissue-specific segmentation of high-grade gliomas in multi-channel mr. In: Proc MICCAI (2012)

Automated Model-Based Segmentation of Brain Tumors in MR Images

Tom Haeck[1], Frederik Maes[1], and Paul Suetens[1]

KU Leuven, Leuven, Belgium

Abstract. We present a novel fully-automated generative brain tumor segmentation method that makes use of a widely available probabilistic brain atlas of white matter, grey matter and cerebrospinal fluid. An Expectation Maximization-approach is used for estimating intensity models for both normal and tumorous tissue. A level-set is iteratively updated to classify voxels as either normal or tumorous, based on which intensity model explains the voxels' intensity the best. No manual initialization of the level-set is needed. The overall performance of the method for segmenting the gross tumor volume is summarized by an average Dice score of 0.68 over all the patient volumes of the BRATS 2015 trainings set.

1 Introduction

Routine use of automated MR brain tumor segmentation methods in clinical practice is hampered by the large variability in shape, size, location and intensity of these tumors. Reviews of MR brain tumor segmentation methods are provided by Bauer et al. [1] and Menze et al. [2].

Brain tumor segmentation methods in Menze et al. [2] are grouped into generative and discriminative methods. Discriminative segmentation methods require a set of manually annotated training images from which the appearance of tumors is implicitly learned by the algorithm. Generative models on the other hand don't require a set of annotated training images. Explicit prior knowledge of anatomy or intensity appearance is directly incorporated into the algorithm [3]. In the past BRATS challenges [2], discriminative methods have largely outperformed generative methods, which sparked increased development in discriminative methods. Although it is clear that existing methods need to be improved in terms of accuracy, the methods also need to be developed and broadened in order to be deployable in clinical settings where access to a training set is limited or non-existent.

We present a novel fully-automated generative tumor segmentation method that only makes use of a widely available probabilistic brain atlas of white matter (WM), grey matter (GM) and cerebrospinal fluid (CSF) and for which no manual initialization is needed. The probabilistic prior is fully exploited by searching globally for voxel intensities that cannot be explained by the normal tissue model. The method is outlined in Sec. 2 and results are presented in Sec. 3.

2 Method

Classification is based on an EM-estimation of normal and tumorous intensity models. An evolving level-set determines which of both intensity models applies to what regions in the image (Fig. 1).

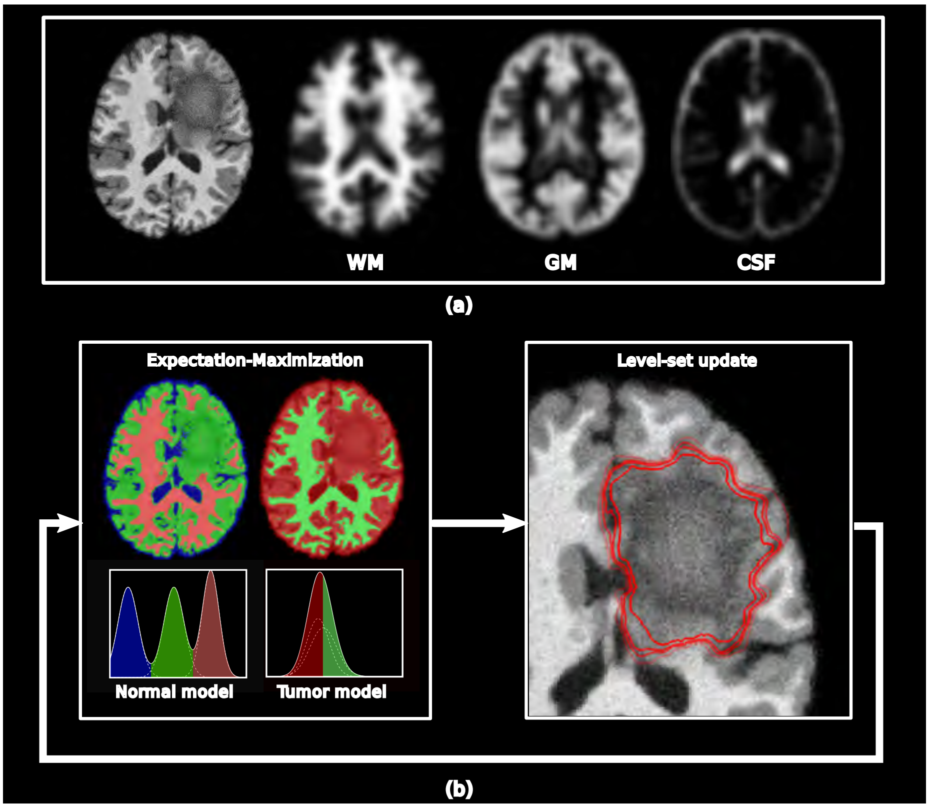


Fig. 1. (a) Spatial priors are non-rigidly registered to the patient image. (b) A full Expectation-Maximization estimation of the normal and tumorous intensity models is done, after which a level-set is updated. This process is repeated until convergence.

Prior Registration Spatial priors of WM, GM and CSF are non-rigidly registered to the patient image. The prior information is relaxed by smoothing the spatial priors with a Gaussian kernel.

Intensity models and the Expectation-Maximization algorithm Normal and tumorous tissue intensities are modeled separately. Let $G_{\mathbf{\Sigma_j}}$ be a zero-mean multivariate Gaussian with covariance matrix $\mathbf{\Sigma_j}$, then normal and tumorous tissue are both modeled by a Gaussian mixture model

$$p(\mathbf{y_i}|\theta) = \sum_{j}^{K} G_{\mathbf{\Sigma_j}}(\mathbf{y_i} - \mu_\mathbf{j}) p(\Gamma_i = j), \tag{1}$$

with $\mathbf{y_i} = (y_{i_1}, \ldots, y_{i_N})$ the intensity of voxel i and $\Gamma_i = \{j | j = 1 \ldots K\}$ the tissue class. The intensity model parameters $\theta = \{(\mu_\mathbf{j}, \mathbf{\Sigma_j}) | j \in 1 \ldots K\}$ are iteratively updated using an EM-approach [3]. For normal tissue, $K = 3$ and $p(\Gamma = j) = \pi_j$ are the spatial priors for WM, GM and CSF. For tumorous tissue, the number of Gaussians is a free parameter and the weights of the Gaussians are updated according to the volume fraction of each of the tumor classes.

Convex level-set formulation The image I is subdivided into two regions Ω_{in} and Ω_{out} for which the intensities are modeled by the probability distributions described in the previous paragraph [4]. The regions are separated by a boundary $\partial\Omega$ that is implicitly represented by a level-set function. The boundary and intensity model parameters are found by minimizing the energy functional

$$\underset{\theta_{in}, \theta_{out}, \partial\Omega}{\operatorname{argmin}} \quad \lambda \int_{\Omega_{in}} -\log p_{in}(I | \Omega_{in}, \theta_{in})\, d\mathbf{x} + \lambda \int_{\Omega_{out}} -\log p_{out}(I | \Omega_{out}, \theta_{out})\, d\mathbf{x} + \kappa \mathrm{L}(\partial\Omega), \tag{2}$$

where $L(.)$ is the length of the boundary. The first two terms penalize the negative loglikelihood of the image I evaluated in respectively the tumorous and normal intensity model. The third term penalizes the length of the boundary. Parameters λ and κ determine the relative importance of the energy terms. For each iteration to update the level-set, a full Expectation-Maximization estimation of the parameters θ_{in} and θ_{out} is done.

The energy functional is non-convex and the gradient flow finds a solution that depends on a manual initialization of the level-set. It is unclear how close the initialization needs to be to the ultimate tumor segmentation. In this work, this problem is overcome by using a convex level-set formulation that performs a global search over the image and makes a manual initialization superfluous. A global minimum is guaranteed by replacing the gradient flow by another gradient flow with the same steady-state solution and by restricting the level-set to lie in a finite interval [5]. The problem is thus reformulated as an L_1-minimization problem that is solved by the Split Bregman-numerical scheme [5]. It is important to note that, by using spatial priors of WM, GM and CSF, the global optimum coincides with the clinically meaningful notion of normal and tumorous regions.

3 Experiments and Results

The method is validated on the BRATS 2015-trainings data set [2] that holds 54 low-grade and 220 high-grade glioma patient volumes that are already skull-stripped and registered intra-patient. No further pre-processing is done. Since the method is designed to segment gross tumor volume, the modalities that are used are the T2-weighted MR image and the T2-weighted FLAIR MR image. The spatial priors are relaxed by a Gaussian kernel with standard deviation of $\sigma = 3$ voxels. The number of Gaussians for modeling the tumor intensities is set to 1. The energy functional hyperparameters are $\lambda = 1e1$ and $\kappa = 1e1$. For each update of the level-set, a full EM-estimation for both the tumorous and normal intensity model is performed. The computation time for a single patient volume is about 15 minutes on a 2×2.66Ghz Quad-Core CPU, out of which 10 minutes are spent for the non-rigid registration of the priors to the patient volume.

The overall average Dice score for the gross tumor volume on the training data set is 0.68. This score is comparable to fully-automated generative methods from the past BRATS challenges that were validated on a data set that is very similar [2]. However, we should note that currently available discriminative algorithms can reach Dice scores of over 0.80.

4 Discussion and Conclusion

In plenty of clinical settings only a handful of patient images needs to be processed without the availability of an annotated training set. Generative methods have therefore an enormous practical value. In this work, we have presented a generative method for segmenting the gross tumor volume in glioma patients. A global search is performed and spatial prior information of healthy human adults is exploited in order to do the segmentation in a fully-automated way.

References

1. S. Bauer, R. Wiest, L.-P Nolte, and M. Reyes. A survey of mri-based medical image analysis for brain tumor studies. *Phys Med Biol*, 58(13):970–129, 2013.
2. B. Menze, M. Reyes, and K. Van Leemput. The multimodal brain tumor image segmentation benchmark (brats). *IEEE Transactions on Medical Imaging*, (99), 2014.
3. K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated Model-Based Tissue Classification of MR Images of the Brain. *IEEE Transactions on Medical Imaging*, 18:897–908, 1999.
4. M. Rousson and R. Deriche. A variational framework for active and adaptative segmentation of vector valued images. In *Proceedings of the Workshop on Motion and Video Computing*, MOTION '02. IEEE Computer Society, 2002.
5. T. Goldstein, X. Bresson, and S. Osher. Geometric applications of the split bregman method: Segmentation and surface reconstruction. *Journal of Scientific Computing*, 45(1-3):272–293, 2010.

A Convolutional Neural Network Approach to Brain Tumor Segmentation

Mohammad Havaei[1], Francis Dutil[1], Chris Pal[2], Hugo Larochelle[1], and Pierre-Marc Jodoin[1]

[1] Université de Sherbrooke, Sherbrooke, Qc, Canada
[2] École Polytechnique de Montréal, Canada

Abstract. We consider the problem of fully automatic brain tumor segmentation, in MR images containing low and high grade glioblastomas. We propose a Convolutional Neural Network (CNN) approach which reaches top performances while also being extremely efficient, a balance that existing methods have struggled to achieve so far. Our CNN is trained directly on the raw image modalities and thus learns a feature representation directly from data. We propose a novel cascaded architecture with two pathways that each model small details in tumors and their larger context. Since the high imbalance of tumor labels can significantly slow down training, we also propose a two-phase, patch-wise training procedure allowing us to train models in a few hours. Fully exploiting the convolutional nature of our model also allows us to segment a complete brain image in 3 minutes. In experiments on the 2013 BRATS challenge dataset, we demonstrate that our approach is among the best performing methods in the literature, while also being very efficient.

1 Introduction

The goal of brain tumor segmentation is to identify areas of the brain whose configuration deviates from normal tissues. Segmentation methods typically look for active tumorous tissues (vascularized or not), necrotic tissues, and edema (swelling near the tumour) by exploiting several Magnetic resonance imaging (MRI) modalities, such as T1, T2, T1-Contrasted (T1C) and Flair.

Recently, Convolutional Neural Networks (CNNs) have proven particularly successful in many computer vision applications. For instance, the so-called AlexNet architecture [7] was the first to establish CNNs as the *de facto* state-of-the-art methodology for object recognition in natural images. The main appeal of convolutional networks is the ability of extracting a deep hierarchy of increasingly complex features. The potential of CNNs for tumor segmentation however is currently poorly understood, and has only been the subject of preliminary investigations (see workshop publications [4, 10, 9]). In other work [6], alternative to the standard CNN framework have also been explored for more general image segmentation tasks, with the argument that CNN training is overly computationally intensive.

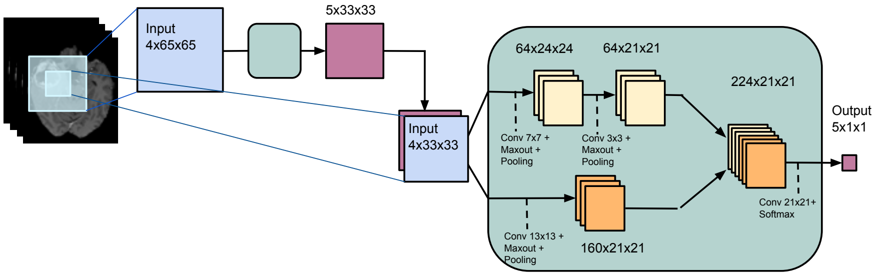


Fig. 1: The InputCascadeCNN model. The input patch goes through two convolutional networks each comprising of a local and a global path. The feature maps in the local and global paths are shown in yellow and orange respectively.

In this paper, we propose a successful and very efficient CNN architecture for brain tumor segmentation. We report results on the MICCAI-BRATS 2013 challenge dataset [1] and confirm that ours is one of the fastest and most accurate approaches currently available.

2 Convolutional Neural Network Architecture

We approach the problem of brain tumor segmentation by solving it slice by slice, from the axial view. Let **X** be one such 2D image (slice), where each pixel is associated with multiple channels, one for each image modality. We treat the problem of segmentation as one of taking any patch it contains and predicting the label of the pixel at its center. The problem is thus converted into an image classification problem.

Figure 1 illustrates our model which we refer to as InputCascadeCNN. As seen from Figure 1, our method uses a two-pathway architecture, in which each pathway is responsible for learning about either the local details or the larger context of tissue appearances (e.g. whether or not it is close to the skull). The pathways are joined by concatenating their feature maps immediately before the output layer.

Finally, a prediction of the class label is made by stacking a final output layer, which is fully convolutional to the last convolutional hidden layer. The number of feature maps in this layer matches the number of class labels and uses the so-called *softmax* non-linearity.

DNN's perform pixel classification without taking into account the local dependencies of labels, one can model label dependencies by considering the pixel-wise probability estimates of an initial CNN as additional input to a second DNN, forming a cascaded architecture.

2.1 Efficient Two-Phase, Patch-Wise Training

By interpreting the output of our CNN as a model for the distribution over segmentation labels, a natural training criteria is to maximize the probability of all labels in our training set or, equivalently, to minimize the negative log-probability $-\log p(\mathbf{Y}|\mathbf{X}) = \sum_{ij} - \log p(Y_{ij}|\mathbf{X})$ for each labeled brain. To do this, we follow a stochastic gradient descent approach by repeatedly selecting labels Y_{ij} at a random subset of positions (i.e. patches) within each brain, computing the average negative log-probabilities for this mini-batch of positions and performing a gradient descent step on the CNNs parameters.

Care must be taken however to ensure efficient training. Indeed, a priori, the distribution of labels is very imbalanced (e.g. most of the brain is non-tumorous). Selecting patches from the true distribution would cause the model to be overwhelmed by healthy patches. It is well known that neural network training algorithms such as stochastic gradient descent perform poorly in cases of strong class imbalances. To avoid these issues, we initially construct our patches dataset such that all labels are equiprobable. This is what we call the *first* training phase. Then, in a *second* phase, we account for the unbalanced nature of the data and re-train only the output layer (i.e. keeping the kernels of all other layers fixed) with a more representative distribution over the labels. Using this approach, we were able to fully train CNNs in less than 6 hours.

3 Implementation details

Our implementation is based on the Pylearn2 which supports GPU's and can greatly accelerate the execution of deep learning algorithms [5].

To test the ability of CNNs to learn useful features from scratch, we employed only minimal preprocessing. We removed the 1% highest and lowest intensities, as done in [8] and applied N4ITK bias correction [3] to T1 and T1C modalities. These choices were found to work best in our experiments. The data was normalized within each input channel, by subtracting the channel mean and dividing by its standard deviation.

The hyper-parameters of the model (kernel and pooling size for each layer) are illustrated in Figure 1. The learning rate α is decreased by a factor $\gamma = 10^{-1}$ at every epoch. The initial learning rate was set to $\alpha = 0.005$.

A post processing method based on connected components was also implemented to remove flat blobs which might appear in the predictions due to bright corners of the brains close to the skull.

4 Experiments and Results

We conducted our experiments on real patient data obtained from the 2013 brain tumor segmentation challenge (BRATS), as part of the MICCAI conference [1]. It contains 20 brains with high grade and 10 brains with low grade tumors for training and 10 brains with high grade tumors for testing. For each brain there

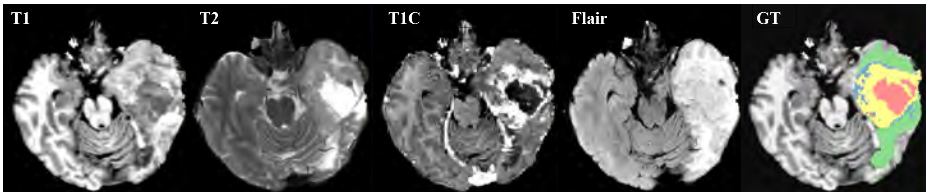


Fig. 2: The four images on the left show the MRI modalities used as input channels to the CNN models and the one on the right shows the ground truth labels, with the following color coding: 🟩 edema, 🟨 enhanced tumor, 🟥 necrosis, 🟦 non-enhanced tumor.

exists 4 modalities, namely T1,T1-Contrasted (T1C), T2 and Flair. The training brains come with a ground truth of 5 segmentation labels, namely *healthy*, *necrosis*, *edema*, *non-enhancing tumor* and *enhancing tumor*. Figure 2 shows an example of the data as well as the ground truth.

Since ground truth segmentations are not available for the test data, a quantitative evaluation of the model is only possible through the BRATS online evaluation system [2]. It reports the *Dice* measure (which is identical to the F score) on three tumor regions, as follows: the *complete* (including all four tumor subclasses), the *core* (including all tumor subclasses except "edema") and the *enhancing* (including the "enhanced tumor" subclass) [8].

The table of Figure 3 shows how our implemented architecture compare to the currently published state-of-the-art methods. The table shows that InputCascadeCNN out performs Tustison et al. the winner of the BRATS 2013 challenge and is ranked first in the table.

Figure 3 shows visual segmentations produced by our model. The larger receptive field in the two-pathway method allows the model to have more contextual information of the tumor and thus yields better segmentations. Also, with its two pathways, the model is flexible enough to recognize the fine details of the tumor as opposed to making very smooth segmentation as in the one path method. By allowing for a second phase training and learning from the true class distribution, the model corrects most of the misclassifications produced in the first phase. Cascading CNNs also helps the model to refine its predictions by introducing label dependencies.

5 Conclusion

In this paper, we proposed a brain tumor segmentation method based on deep convolutional neural networks. Our method is among the most accurate methods available, while being the most efficient. The high performance is achieved with the help of a novel two-pathway architecture (which can model both the local details and global context) as well as modeling local label dependencies by stacking two CNN's

Method	Dice (F1)		
	Complete	Core	Enhancing
InputCascadeCNN	0.88	0.79	0.73
Tustison[8]	0.87	0.78	0.74
Meier[8]	0.82	0.73	0.69
Reza[8]	0.83	0.72	0.72
Uhlich[8]	0.83	0.69	0.68
Zhao[8]	0.84	0.70	0.65
Cordier[8]	0.84	0.68	0.65
Festa[8]	0.72	0.66	0.67
Doyle[8]	0.71	0.46	0.52

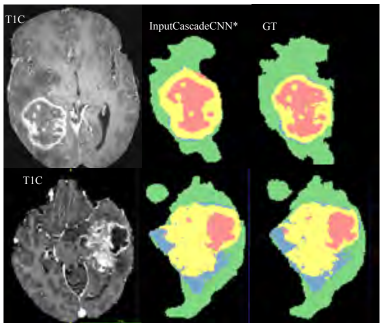


Fig. 3: The table compares the results of our 4 architectures with the state-of-the-art methods on the BRATS-2013 testset. The images show the segmentations predicted by our methods and the corresponding ground truth with the following color code: 🟩 edema, 🟨 enhanced tumor, 🟥 necrosis, 🟦 non-enhanced tumor.

References

1. Brats 2014 challenge manuscripts. http://www.braintumorsegmentation.org
2. Virtual skeleton database. http://www.virtualskeleton.ch/
3. Avants, B.B., et al.: Advanced normalization tools (ants). Insight J (2009)
4. Davy, A., et al.: Brain tumor segmentation with deep neural networks. proc of BRATS-MICCAI (2014)
5. Goodfellow, I., et al.: Pylearn2: a machine learning research library. arXiv preprint arXiv:1308.4214 (2013)
6. Huang, G.B., Jain, V.: Deep and wide multiscale recursive networks for robust image labeling. ICLR, arXiv:1310.0354 (2014)
7. Krizhevsky, A., et al.: ImageNet classification with deep convolutional neural networks. In: NIPS (2012)
8. Menze, B., et al: The multimodal brain tumor image segmentation benchmark (brats). Medical Imaging (2014)
9. Urban, G., et al.: Multi-modal brain tumor segmentation using deep convolutional neural networks. proc of BRATS-MICCAI (2014)
10. Zikic, D., et al.: Segmentation of brain tumor tissues with convolutional neural networks. proc of BRATS-MICCAI (2014)

Multimodal Brain Tumor Segmentation (BRATS) Using Sparse Coding and 2-layer Neural Network

Assaf Hoogi, Andrew Lee, Vivek Bharadwaj, and Daniel L. Rubin,

Department of Radiology and Medicine (Biomedical Informatics Research), Stanford University School of Medicine, CA, USA.

1. MATERIALS AND METHODS

1.1 Dataset description

Due to computational load, for our initial analysis we used 100 MRI scans, from the total of 220 that were supplied. The data set contains high- and low- grade glioma cases that were scanned by 4 different modalities - T1 MRI, T1 contrast-enhanced MRI, T2 MRI, and T2 FLAIR MRI. Each scan is a 3D volume that includes 155 2D slices. Each scan can contain one or more of the following - normal tissue, necrosis, edema, non-enhancing tumor, and enhancing tumor. All data sets have been aligned to the same anatomical template and interpolated to 1mm^3 voxel resolution. Annotations comprise the “whole” tumor, the tumor “core” (including cystic areas), and the Gd-“enhanced tumor core” [1]. The ground truth has been supplied by BRATS challenge and was approved by experience observers. It includes separate binary mask for each intra-lesion region (‘ground-rule mask’).

1.2 Preprocessing

The initial step is to do gray level normalization, by creating a mask of all pixels in the brain that represents the average gray value of brain pixels and which is subtracted from each pixel in the brain image. Our segmentation includes several steps - feature extraction, feature representation and classification (fig. 1).

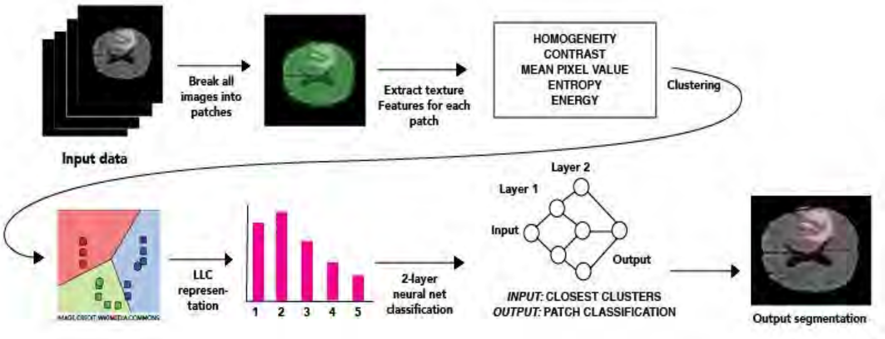

Input data
Break all images into patches
Extract texture Features for each patch
HOMOGENEITY
CONTRAST
MEAN PIXEL VALUE
ENTROPY
ENERGY
Clustering
IMAGE CREDIT: WIKIMEDIA COMMONS
LLC represen-tation
1
2
3
4
5
2-layer neural net classification
Layer 2
Layer 1
Input
Output
INPUT: CLOSEST CLUSTERS
OUTPUT: PATCH CLASSIFICATION
Output segmentation

Figure 1: Patch classification algorithm

1.3 Feature extraction

The first step is feature extraction. For this process, we first extract 5*5 patches from each brain mask that is created during the preprocessing. We use patch size that is larger enough to capture high-level semantics such as edges or corners. At the same time, the patch size should not be too large if it is aimed to serve as a common building block for many images. Once patches are defined, each patch is represented with a set of feature descriptors. In our implementation we use the mean grey level, in addition to 4 additional Haralick features (homogeneity, contrast, entropy, and energy) based on their ability to best represent the rough and fine spatial information. Haralick texture features are extracted from a second order statistics model, Gray-Level Co-occurrence Matrices (GLCM) [2]. One-pixel distance between examined pixels and 4 different angular directions θ are used (0, 90, 180, 270). Normalization of the GLCM is done, thus the sum of its elements is equal to 1. PCA was applied in order to reduce features dimensionality - 5 PCA components were chosen [3].

1.4 Dictionary reconstruction

In our method, we use 4 different dictionaries. Every dictionary is constructed for one specific modality, taking into account the variability of the cases screened by this modality. After obtaining the features vectors from all patches, they are clustered by using the k-means algorithm. The centers of the centroids are not chosen randomly; rather, they are chosen so they will have to be as far as they can from each other. This will ensure the stability of the clustering procedure. We use 200 clusters for the k-means. In the presented method, we build the dictionary by using Locality-constrained Linear Coding (LLC) method [4], a sparse coding method that enables the clustering of each feature vector to the 5 closest clusters, not only to the single closest one. By using LLC, the loss of spatial information is minimized. This is important especially when a feature vector is near several clusters by using Euclidean distance, rather than being close to a dominant one.

1.5 Classification

A feed-forward neural net [5] is then trained to predict the label of a patch (using majority voting in the ground-rule mask), given the vector of closest clusters. For classification, we use a neural network method consisting of an input layer for the 5 closest-clusters vector, 2 hidden layers with 100 neurons each, and an output layer that predicts the patch label. Every neuron in a hidden layer of the neural network receives input from every neuron in the previous layer. The output of a neuron is determined by computing the net input of the neuron and feeding it into a transfer function. The input weights for the neurons are initialized to random values and the neural net is the trained using Levenberg-Marquardt backpropagation. The network is trained until the gradient drops below 10^{-2}.

1.6 Evaluation

The results have been evaluated by comparing them to the supplied ground truth. Two-fold cross validation was done by dividing the dataset into 2 subsets, each contains 50 3D cases. We first trained on one subset of 50 cases and tested on the second one, then we switched the sets. We calculated the Dice coefficient between the mask of each label (normal tissue, necrosis, edema, non-enhancing tumor, and enhancing tumor) and the one that was obtained by our automated method.

2. RESULTS

Figure 2 shows few examples for the detection and the segmentation of the brain tumors and their internal sub-regions. The Dice criterion was calculated separately for each 2D slice. The median value of those Dice scores was 0.7315 for the "whole" lesion, 0.6347 for the "core" of the lesion and 0.8359 for the "active" part, as were defined in [1].

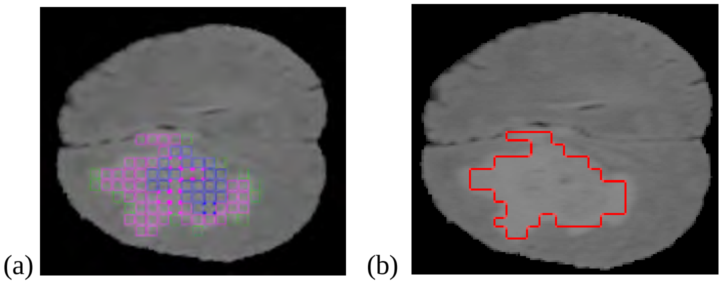


Figure 2: Examples for lesion detection and classification. The different colors in solid correspond to different sub-regions in the ground truth mask that are correctly classified. Colored stars refers to labels misclassifications. Dotted green squares refer to false negative. (a) the detection of the lesion and its internal sub-regions. (b) the obtained segmentation – according to (a)

3. DISCUSSION

Our method shows novel integration of LLC representation and Neural Network classification to detect brain tumors and distinguish between its internal parts of brain lesions. It shows promising results for detecting and segmenting those regions. However, the method can still be improved, especially for low contrast lesions. Further work will include larger dataset to evaluate the accuracy and the stability of its performance and refinement of the method by using additional features

3. REFERENCES

[1] Menze B, Jakab A, Bauer S, Kalpathy-Cramer J, Farahani K, Kirby J, Burren Y, Porz N, Slotboom J, Wiest R, Lanczi L, Gerstner E, Weber M-A, Arbel T, Avants B, Ayache N, Buendia P, Collins L, Cordier N, Corso J, Criminisi A, Das T, Delingette H, Demiralp C, Durst C, Dojat M, Doyle S, Festa J, Forbes F, Geremia E, Glocker B, Golland P, Guo X, Hamamci A, Iftekharuddin K, Jena R, John N, Konukoglu E, Lashkari D, Antonio Mariz J, Meier R, Pereira S, Precup D, Price SJ, Riklin-Raviv T, Reza S, Ryan M, Schwartz L, Shin H-C, Shotton J, Silva C, Sousa N, Subbanna N, Szekely G, Taylor T, Thomas O, Tustison N, Unal G, Vasseur F, Wintermark M, Hye Ye D, Zhao L, Zhao B, Zikic D, Prastawa M, Reyes M, Van Leemput K. The multimodal brain tumor image segmentation benchmark (brats). IEEE Trans Med Imaging 2014

[2] Haralick, R.M., Shanmugam K. and Dinstein I., Textural Features for Image Classification, IEEE Transactions on Systems, Man and Cybernetics, SMC vol. 3, no. 6, pp. 610-620, 1973.

[3] H. P. Kriegel, P. Kröger, E. Schubert, A. Zimek, "A General Framework for Increasing the Robustness of PCA-Based Correlation Clustering Algorithms," in Proceedings of the 20th international conference on Scientific and Statistical Database Management (SSDBM), edited by B. Ludascher and N. Mamoulis (Springer-Verlag Berlin, Heidelberg, 2008), pp. 418-435.

[4] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, 2010, pp. 3360-3367.

[5] S. Haykin, Neural Networks: A Comprehensive Foundation, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.

Highly discriminative features for glioma segmentation in MR volumes with random forests

Oskar Maier[1,2], Matthias Wilms[1], and Heinz Handels[1]

[1] Institute of Medical Informatics, Universität zu Lübeck
[2] Graduate School for Computing in Medicine and Life Sciences, Universität zu Lübeck
maier@imi.uni-luebeck.de

Abstract. Automatic segmentation of brain tumors is necessary for standardized, reproducible and reliable procedures in diagnosis, assessment and management. This article details a contribution to the Multimodal Brain Tumor Image Segmentation Benchmark (BRATS) organized in conjunction with the MICCAI 2015. The proposed method bases on decision forests trained on a set of dedicated features carefully selected for their ability to discriminate pathological from normal tissue in brain MRI volumes. The method is described in detail and all chosen parameter values are disclosed. Preliminary results on the training data places the approach among the highest ranking contributions.

Keywords: brain tumor, high grade glioma, low grad glioma, magnetic resonance imaging, MRI, random forest, RDF

1 Introduction

Gliomas are a type of tumor originating from glial cells, usually found in the brain or the spine. They can be categorized according to their World Health Organization (WHO) severity grade into Low Grade Gliomas (LGG) and High Grade Gliomas (HGG), where the former are well-differentiated and the latter not. Since gliomas make up 80% of all malignant brain tumors, the relative survival rate is low. Available treatment options are often aggressive, such as e.g. surgery, radiation therapy and chemotherapy. Diagnosis, assessment and treatment planing include the use of intensive neuroimaging protocols to evaluate disease progression, location, type and treatment success. In clinical routine, only rudimentary quantitative assessment methods are employed up to date, if at all. To standardize procedures and ensure high quality, it would be highly desirable to introduce automatic, robust, reliable and reproducible automatic segmentation methods for glioma. Previous challenges have shown the task to be demanding and no satisfying solution has been found yet [4].

2 Method

The challenge's training data consists of multi-spectral (T1, T1c, T2, Flair) scans of 274 patients, some with LGGs, others with HGGs. The provided ground-truth of some cases has been created manually, but for the majority a fusion of high ranking methods from previous versions of the challenge has been employed. The testing data will only include cases with expert created ground-truth.

2.1 Pre-processing

The image data is provided with a 1 mm isotropic resolution, already co-registered, skull-stripped and registered to a template image. Nevertheless, the training cases of the challenge display high intensity differences, a normal occurrence for MRI, where intensity ranges are not standardized. With a learning based intensity standardization method implemented in MedPy [2] and based on [5] we harmonize each sequences intensity profile.

2.2 Forest classifier

We employ the random forest (RF) classifier implemented in [6], which is similar to the propositions made by [1]. The classification of brain lesions in MRI is a complex task with high levels of noise [3], hence a sufficiently large number of trees must be trained.

2.3 Features

The primary distinction criteria for identifying pathological tissue of gliomas is the MR intensity in the different sequences. The bulk of our voxel-wise features therefore bases on the intensity values.

intensity First feature is the voxel's intensity value.

gaussian Due to the often low signal-to-noise ratio in MR scans and intensity inhomogeneities of the tissue types, we furthermore regard each voxel's value after a smoothing of the volume with a 3D Gaussian kernel at three sizes: $\sigma = 3, 5, 7\ mm$.

hemispheric difference Gliomas mostly affect a single hemisphere, therefore we extract the hemispheric difference (in intensities) after a Gaussian smoothing of $\sigma = 1, 3, 5\ mm$ to account for noise. Since the volumes are provided already registered to a template image, the central line of the saggital view is taken as sufficiently close approximation of the sagittal midline.

local histogram Another employed feature is the local histogram, as proposed in [3], which provides information about the intensity distribution in a small neighbourhood around each voxel. The neighbourhoods considered were $R = 5^3, 10^3, 15^3\ mm$, the histogram was fixed to 11 bins.

center distance Finally, we extract the distance to the image center (assumed here to coincide roughly with the brain's center of mass) in *mm* as final feature. Note that this is not intensity based, but rather discloses each voxel's rough location inside the brain.

All features are extracted from each of the MR sequence, hence in total we obtain 163 values per multi-spectral voxel. Note that all of the employed features are implemented in MedPy [2].

3 Experiments

3.1 Training choices and parameter values

For training our RF, we sample $1,000,000$ voxels randomly from all training cases. The ratios between classes in each case are largely kept intact (i.e. tumor class samples will be under-represented), but the minimum of samples drawn for each class from each case is set to 50. A total of 100 trees are trained for the forest. As split criteria the Gini impurity is employed, a maximum of $\sqrt{163}$ features is considered at each node. No growth restrictions are imposed.

3.2 Preliminary results

Independent online evaluation is provided by the challenge organizers for (a) the complete glioma, (b) the core and (c) the enhancing tumor as region of special interest. Employed measures are Dice's coefficient (DC), the positive predictive value (PPV) and sensitivity (SE). Using a leave-one-out evaluation scheme, we have obtained the scores presented in Tab. 1 on the 55 LGG cases.

Table 1. Evaluation results on 55 LGG training cases. See the text for details on the abbreviations employed.

Complete			Core			Enhancing		
DC	PPV	SE	DC	PPV	SE	DC	PPV	SE
0.84	0.84	0.85	0.66	0.70	0.72	0.39	0.47	0.43

4 Discussion and conclusion

We have shown our proposed method to be a suitable approach for glioma segmentation in brain MR volumes, with high overall DC values. In the case of the enhancing tumor our approach shows need for improvement.

RF are fast and robust ensemble classifiers which already have been shown to be suitable for other brain pathology segmentation tasks [3]. They are easy to train and give consistent results for a large range of parameters.

On the downside, they suffer from the same drawbacks as all other machine learning based methods: The training set must be carefully chosen and types of cases not present in the training data can not be processed.

References

1. Criminisi, A., Shotton, J., Konukoglu, E.: Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. Foundations and Trends® in Computer Graphics and Vision 7(2–3), 81–227 (2012)
2. Maier, O.: MedPy. https://pypi.python.org/pypi/MedPy, accessed: 2015-03-29
3. Maier, O., Wilms, M., et al.: Extra tree forests for sub-acute ischemic stroke lesion segmentation in MR sequences. Journal of Neuroscience Methods 240(0), 89–100 (2015)
4. Menze, B., Reyes, M., Van Leemput, K.: The Multimodal Brain TumorImage Segmentation Benchmark (BRATS). Medical Imaging, IEEE Transactions on PP(99), 1–1 (2014)
5. Nyul, L., Udupa, J., Zhang, X.: New variants of a method of mri scale standardization. Medical Imaging, IEEE Transactions on 19(2), 143–150 (Feb 2000)
6. Pedregosa, F., Varoquaux, G., et al.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)

CaBS: A Cascaded Brain Tumor Segmentation Approach

Eric Malmi[1,2], Shameem Parambath[2], Jean-Marc Peyrat[3], Julien Abinahed[3], and Sanjay Chawla[2]

[1] Aalto University, Espoo, Finland
[2] Qatar Computing Research Institute, Doha, Qatar
[3] Qatar Robotic Surgery Centre, Qatar Science and Technology Park, Doha, Qatar

Abstract. We propose a cascaded workflow approach to carry out the brain tumor segmentation task on images provided as part of MICCAI 2015 challenge. After the necessary data normalization and feature generation task, we first apply a random forest classifier to distinguish brain tissue into tumor and non-tumor regions. A post-processing step is then carried out to extract large connected components of the tumor tissue. A second level of classification, again using random forests, is then performed to distinguish between different tumor types. Our workflow is flexible to incorporate different types of classifiers and pre-processing and post-processing strategies.

1 Introduction

In this paper we propose a cascaded brain tumor segmentation approach (CaBS) to identify tumor from brain MRI scan images. The advantage of our approach is that each step in the workflow can be independently tuned and optimized to create a reliable segmentation engine which can evolve over time as more training data becomes available. The design of the proposed workflow is shown in Figure 1. The steps of the workflow include: (i) pre-processing of data, (ii) feature engineering, (iii) a coarse level classification to distinguish tumor and non-tumor regions, (iv) a morphological operation to form connected components of the tumor region (iv) a finer level spatially regularized classification to distinguish Necrosis, Edema, Non-Enchancing and Enhancing Tumor and (v) result preparation. In the rest of the paper we provide details of each of these steps and showcase our results on MICCAI 2015 challenge.

2 Pre-Processing & Feature Extraction

2.1 Pre-Processing of Images

First of all, we used N4ITK in ANTS [9,8] on T1 and T1c images for MRI bias field correction. We did not use it on FLAIR and T2 images since this correction seemed to lower the tumor contrast as also noticed in [2]. We created a brain

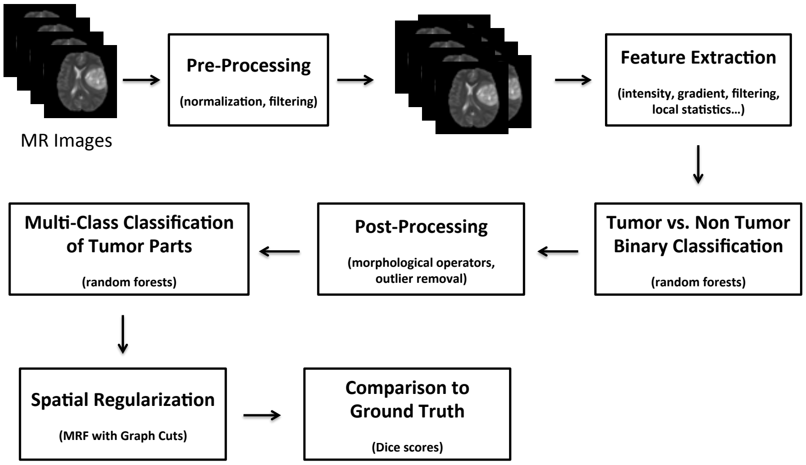

MR Images
Pre-Processing
(normalization, filtering)
Feature Extraction
(intensity, gradient, filtering, local statistics...)
Tumor vs. Non Tumor Binary Classification
(random forests)
Post-Processing
(morphological operators, outlier removal)
Multi-Class Classification of Tumor Parts
(random forests)
Spatial Regularization
(MRF with Graph Cuts)
Comparison to Ground Truth
(Dice scores)

Fig. 1: The proposed brain tumor segmentation workflow (CaBS)

mask from the FLAIR image of each patient including the voxels with strictly positive values. In the remainder, further processing was limited to this brain mask. Finally, we normalized intensities of each image such that the mode of the histogram is aligned to 0 and such that the standard deviation with respect to the mode in its neighborhood (±2 standard deviation of the whole histogram) value is equal to 1. We also created additional smoothed version of these images (Gaussian filter with $\sigma = 5mm$).

2.2 Computation of Features

For each of the resulting 8 images, we compute the following 10 features at each voxel: intensity, gradient magnitude (Sobel filter), laplacian ($\alpha = 0.5$), as well as standard deviation, range, entropy, skewness, kurtosis, minimum and maximum in a 5 voxel neighborhood. We obtain a total of 80 features per voxel and patient.

3 Classification Methodology

We use a two-step learning approach to classify the tumor tissues. In the first step, we learn a classifier to differentiate between the non-tumor tissues (voxels with label zero) and tumor tissues (voxels with non-zero labels). The second step sub-classifies the tumor tissues to four different sub-categories.

The motivation behind the two-step approach is the following

- Since a majority of the voxels comes under label zero, a two-step classification makes the problem more balanced by running a binary classifier in the first stage and a multi-class classifier in the second stage.
- Theoretical properties of Dice score as a performance measure for class-imbalanced classification is not yet known (In general F-score is recommended for class-imbalanced classification [6]), and by making the problem more balanced we can guarantee the statistical consistancy and generalization error bounds of the widely used classification algorithms.
- Two-step classification allows us to carry out spatial post-processing to refine the tumor area before separating different tumor types.
- The results we obtained by a two-step classifier are better than a single-step multi-class classifier (see Section 4.2).

We carry out post-processing of the classifier outputs to improve our final Dice scores. In this section we detail the methodology and algorithms for the classification and the post-processing tasks. The code for the proposed approach is available at: `https://github.com/ekQ/brain-tumor-segmentation`

3.1 Tumor vs. Non-Tumor Classification

In the first level of classification, we train a classifier to demarcate tumor tissues and non-tumor tissues. Following the footsteps of past years submissions, we used an ensemble classifier, random forest [1], to carry out the first level of classification. Additionally, we use a thresholding strategy to identify the tumor area. Instead of labeling all voxels with tumor probability above 0.5 as tumor, we optimize this threshold to maximize the complete Dice score on a separate validation set. A threshold of 0.60 was found to be optimal in our experiments.

3.2 Multi-Class Tumor Classification

Second step in our pipeline is the multi-class classification of different tumor labels. The dataset contains four types of tumor, namely *(i)* label 1 for necrosis *(ii)* label 2 for edema *(iii)* label 3 for non-enhancing tumor, and *(iv)* label 4 for enhancing tumor. We use the same training set as used in the first step, but filter out the voxels with label zero. The multi-class classification is carried out using a random forest classifier.

In addition, we tested the second step classification using Kernel SVM with an RBF kernel, but this method did not scale well enough. We also tried running the multi-class classification twice, feeding the first run label probability estimates of the voxel and its neighboring voxels as an input for the second run. Nonetheless, the performance gain was negligible.

3.3 Post-Processing

We carry out post-processing in multiple steps: after running the first-stage classifier and after running the second-stage classifier. In the first post-processing step, we use the “closing” operation and connected component removal. The “closing” operation is widely used in image analysis tasks to remove “salt & pepper” [7]. Closing comprises of two operations, dilation followed by erosion. In dilation, value of a voxel in a given co-ordinate is set to the maximum over all the voxels in the neighborhood, defined by a closed ball centered at the co-ordinate. Erosion is the opposite of dilation. In erosion, value of a voxel in a given co-ordinate is set to the minimum over all the voxels in the neighborhood, defined by a closed ball centered at the co-ordinate.

In addition, we carry out connected component removal to smooth the images. We find the connected component in the image, viewing it as a graph where each voxel is represented as a node connected to its 26 neighboring voxels. We then remove the connected components with less than 3 000 voxels as was done in [4].

In the second post-processing step, we use Markov Random Fields (MRFs) to smooth different tumor regions. MRFs are only applied to the tumor region. Quadratic costs are used for penalizing adjacent tumor voxels with different labels, assuming an ordering of different tumor regions. The following ordering, based on a visual inspection of the ground truth data, is used: edema, necrosis, non-enhancing, and enhancing tumor.

4 Results

4.1 Data

The BRATS Challenge dataset[4][3,5] includes 274 patients (220 high grade tumors and 54 low grade tumors) with 4 imaging modalities (FLAIR, T1, T1c, T2) for each patient. Images were all provided resampled at the same resolution of $1 \times 1 \times 1\ mm^3$ and dimension of $240 \times 240 \times 155$ voxels. We used the complete dataset at full resolution.

4.2 One-Stage vs. Two-Stage Classification

To evaluate the effectiveness of the two-stage classification approach, we compared it with a standard random forest classifier with five classes. The models were trained using 10 000 randomly sampled voxels from 50 train patients and tested on all voxels from 50 test patients. The patients were randomly split into train and test and the same split was used for both methods.

The results are shown in Table 1. The results suggest that the two-stage approach improves the Dice scores at least for core and enchancing tumor. However, a more comprehensive experiment including statistical significance measures should be conducted in order to confirm this observation.

[4] https://www.virtualskeleton.ch/BRATS/Start2015

Table 1: Comparing a standard (one-stage) random forest classifier with the proposed two-stage random forest classifier.

Method	Whole	Core	Enhancing
One-Stage	0.806	0.695	0.621
Two-Stage	0.807	0.710	0.638

4.3 Validation

The following setup is used for producing the predictions for the training data. Use two-fold cross-validation but take only 80 patients from the training fold to reduce memory usage. From each training patient, 100 000 randomly sampled voxels are selected and for the test patients, all voxels are used. The number of trees in the two random forest models is set to 64, the radius of closing to 6, and the tumor threshold in the first stage to 0.6.

The following Dice score values were obtained using this setup: 0.82 for the complete tumor, 0.67 for tumor core, and 0.68 for enchancing tumor.

5 Conclusions

We present a novel approach for classifying tumor cells. Our approach differs from past year's submissions, as we allow more flexibility by cascading different classification approaches together. One major aspect of our approach is that the system allows processing of the intermediate classification results. By employing simple morphological changes, we can fine tune the intermediate results and feed to the final classifier. Our empirical results are close to the top performing methods, and leaves room for improvement by using more sophisticated classification scheme and intermediate processing of the results.

References

1. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
2. Davy, A., Havaei, M., Warde-Farley, D., Briard, A., Tran, L., Jodoin, P.M., Courville, A., Larochelle, H., Pal, C., Bengio, Y.: Brain Tumor Segmentation with Deep Neural Networks. In: MICCAI - BRATS (2014)
3. Kistler, M., Bonaretti, S., Pfahrer, M., Niklaus, R., Büchler, P.: The Virtual Skeleton Database: An Open Access Repository for Biomedical Research and Collaboration. Journal of Medical Internet Research 15(11), e245 (November 2013), http://www.jmir.org/2013/11/e245/
4. Kleesiek, J., Biller, A., Urban, G., Köthe, U., Bendszus, M., Hamprecht, F.A.: *ilastik* for Multi-modal Brain Tumor Segmentation (2014)
5. Menze, B., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M.A., Arbel, T., Avants, B., Ayache, N., Buendia, P., Collins, L., Cordier, N., Corso, J., Criminisi, A., Das, T., Delingette, H., Demiralp, C., Durst, C., Dojat,

M., Doyle, S., Festa, J., Forbes, F., Geremia, E., Glocker, B., Golland, P., Guo, X., Hamamci, A., Iftekharuddin, K., Jena, R., John, N., Konukoglu, E., Lashkari, D., Antonio Mariz, J., Meier, R., Pereira, S., Precup, D., Price, S.J., Riklin-Raviv, T., Reza, S., Ryan, M., Schwartz, L., Shin, H.C., Shotton, J., Silva, C., Sousa, N., Subbanna, N., Szekely, G., Taylor, T., Thomas, O., Tustison, N., Unal, G., Vasseur, F., Wintermark, M., Hye Ye, D., Zhao, L., Zhao, B., Zikic, D., Prastawa, M., Reyes, M., Van Leemput, K.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Transactions on Medical Imaging p. 33 (2014), https://hal.inria.fr/hal-00935640

6. Parambath, S.P., Usunier, N., Grandvalet, Y.: Optimizing f-measures by cost-sensitive classification. In: Advances in Neural Information Processing Systems. pp. 2123–2131 (2014)
7. Serra, J.: Image analysis and mathematical morphology: Theoretical advances. Image Analysis and Mathematical Morphology, Academic Press (1988)
8. Tustison, N., Avants, B., Cook, P., Zheng, Y., Egan, A., Yushkevich, P., Gee, J.: N4ITK: Improved N3 Bias Correction. IEEE Transactions on Medical Imaging 29(6), 1310–1320 (June 2010)
9. Tustison, N., Gee, J.: N4ITK: Nick's N3 ITK Implementation for MRI Bias Field Correction. IThe Insight Journal (2009)

Parameter Learning for CRF-based Tissue Segmentation of Brain Tumors

Raphael Meier[1], Venetia Karamitsou[1], Simon Habegger[2], Roland Wiest[2], and Mauricio Reyes[1]

[1] Institute for Surgical Technologies and Biomechanics, University of Bern
[2] Inselspital, Bern University Hospital, Switzerland
raphael.meier@istb.unibe.ch

Abstract. In this work, we investigate the potential of a recently proposed parameter learning algorithm for Conditional Random Fields (CRFs). Parameters of a pairwise CRF are estimated via a stochastic subgradient descent of a max-margin learning problem. We compared the performance of our brain tumor segmentation method using parameter learning to a version using hand-tuned parameters. Preliminary results on a subset of the BRATS2015 training set show that parameter learning leads to comparable or even improved performance. Future work will include training on the complete data set and the use of more elaborate loss functions suitable for brain tumor segmentation.

1 Introduction

Brain tumor segmentation yields information about the volume of a tumor and its position relative to neighboring possibly eloquent brain areas. Alternatively, such information can only be obtained via time-consuming and subjective manual segmentation. Consequently, fully-automatic segmentation methods applicable in a wide range of domains such as neurooncology, neurosurgery and radiotherapy are in high demand.

The development of new brain tumor segmentation methods has been fostered through the MICCAI Brain Tumor Segmentation (BRATS) Challenge [4], which was held for the first time during MICCAI 2012. Several previously published segmentation methods rely on the use of structured prediction including approaches such as Markov or Conditional Random Fields (CRFs) (e.g. [7, 3]). However, parameters for those models are often hand-tuned rather than estimated from training data. Recently, an efficient method for parameter learning in CRFs applicable to volumetric imaging data was proposed [2]. In this paper, we investigate a modification of our previous segmentation method [3] employing the learning algorithm of [2].

2 Methods

Our current segmentation method (proposed in [3]) encompasses a preprocessing, a feature extraction step followed by a voxel-wise classification and a spatial regularization. The features try to capture visual cues of appearance and image context relevant for discriminating the different tissue classes. Classification is performed by a decision forest. Spatial regularization is formulated as an energy-minimization problem of a CRF. In the remainder of this paper, we present a modification of the spatial regularization used so far.

Structural MRI. Our approach relies on four different MRI sequences, namely T_1-, T_1-post contrast-, T_2-, $FLAIR$-weighted images. We assume that these images are co-registered and organized as a vector image, where every voxel contains the four different MR intensity values. We refer to this image as $X = \left\{\mathbf{x}^{(i)}\right\}_{i \in V}$, where voxel i is represented by a feature vector $\mathbf{x}^{(i)} \in \mathbb{R}^4$ and V denotes the set of all voxels in X. The corresponding tissue label map of X is denoted by $Y = \left\{y^{(i)}\right\}_{i \in V}$ with $y^{(i)}$ being a scalar tissue label (e.g. 1=necrosis, 2=edema, etc.). We consider seven possible tissue classes ($|\mathcal{L}|$=7): three unaffected (gray matter, white matter, csf) and four tumor tissues (necrosis, edema, enhancing and non-enhancing tumor). All possible labelings are contained in $\mathcal{Y}$.

Conditional Random Field. A CRF models a parametrized conditional probability $p\left(Y|X, \mathbf{w}\right) = \frac{1}{Z(X,\mathbf{w})} \exp\left(-E(X, Y, \mathbf{w})\right)$ where $Z(X, w)$ is the partition function. The energy $E(X, Y, \mathbf{w})$ depends linearly on the unknown parameters $\mathbf{w}$. In general, given the parameter vector $\mathbf{w}$, a CRF can predict the labeling Y of a given input image X by minimizing the energy, i.e. $Y^\star = \arg\min_{Y \in \mathcal{Y}} E(X, Y, \mathbf{w})$.

Energy Function. We employ an energy function associated with a pairwise CRF: $E(X, Y, \mathbf{w}) = \sum_{i \in V} D_i(\mathbf{x}^{(i)}, y^{(i)}) + \sum_{(i,j) \in E} B_{i,j}(\mathbf{x}^{(i)}, y^{(i)}, \mathbf{x}^{(j)}, y^{(j)})$. The unary potentials D_i and pairwise potentials $B_{i,j}$ are expressible as an inner product between the parameter vector $\mathbf{w}$ and a feature map ψ_i or $\psi_{i,j}$, respectively [2]. For a given feature vector $\mathbf{x}^{(i)}$, we can define the feature map $\psi_i = \left[I(y^{(i)} = 1)(-\log(p(y^{(i)} = 1|\mathbf{x}^{(i)}))), \cdots, I(y^{(i)} = 7)(-\log(p(y^{(i)} = 7|\mathbf{x}^{(i)})))\right]^T$ by using the indicator function I (returns a value of 1 if the argument is true). The posterior probability $p(y^{(i)}|\mathbf{x}^{(i)})$ is output by the decision forest classifier. Consequently, the cost of assigning label y to voxel i is smaller the more confident the prediction of the decision forest is. The pairwise feature map is given by $\psi_{i,j} = \left[I(y^{(i)} = a, y^{(i)} = b)(1 - I(y^{(i)} = y^{(j)})) \exp\left(-\left\|\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\right\|_\infty\right)\right]_{(a,b) \in \mathcal{L}^2}$ which is defined for all possible label pairs in $\mathcal{L}$. The term $1 - I(y^{(i)} = y^{(j)})$ establishes a Potts-like model. The exponential term penalizes large intensity discontinuities between neighboring voxels. Potentials can now be expressed as an inner product between parameter vector and feature map, i.e. $\langle \mathbf{w}, \psi \rangle$. Furthermore, let $\varPsi^D = \sum_{i \in V} \psi_i$ and $\varPsi^B = \sum_{(i,j) \in E} \psi_{i,j}$. Given the parameter vector $\mathbf{w} = \left[(\mathbf{w}^D)^T, (\mathbf{w}^B)^T\right]^T$, the energy function can then be rewritten as $E(X, Y, \mathbf{w}) = \left\langle \mathbf{w}^D, \varPsi^D \right\rangle + \left\langle \mathbf{w}^B, \varPsi^B \right\rangle$.

Parameter Learning. For estimating the parameter vector $\mathbf{w}$, we use the recently proposed method by Lucchi et al. [2] which builds on the max-margin formulation for parameter learning [6]. Essentially, learning is posed as a quadratic program with soft margin constraints. The objective function is minimized via stochastic subgradient descent in which iteratively a training example $\left(X^{(n)}, Y^{(n)}\right)$ is chosen, the subgradient with respect to this example computed and the weight vector updated accordingly (see algorithm 1). The objective function for $\left(X^{(n)}, Y^{(n)}\right)$ is defined as $f(\mathbf{w}, n) = l\left(Y^{(n)}, Y^{\star}, \mathbf{w}\right) + \frac{1}{2C}\left\|\mathbf{w}\right\|^2$ with l being the hinge loss[3]. The task-specific loss is defined as $\Delta\left(Y^{(n)}, Y\right) = \sum_{i \in V} I\left(y^{(i)} \neq y^{(n),(i)}\right)$ and measures the dissimilarity between a labeling Y and its ground truth $Y^{(n)}$. In contrast to [5], the method of Lucchi et al. aims at an increased reliability in the computation of the subgradient by the use of working sets of constraints $\mathcal{A}^n$. For every iteration, loss-augmented inference is performed to obtain a current estimate of the labeling $Y^{\star} = \arg\min_{Y \in \mathcal{Y}} \left(E(X, Y, \mathbf{w}) - \Delta\left(Y^{(n)}, Y\right)\right)$ (step 4). The set $\mathcal{A}^{n'}$ contains all labelings (constraints) Y which are violated (i.e. $l(Y, Y^{(n)}, \mathbf{w}) > 0$) (step 7). The subgradient is then computed as an average subgradient over all violated constraints (step 8).

Algorithm 1 Subgradient Method with Working Sets [2]

```
1: Training data $\mathcal{S} = \left\{(X^{(i)}, Y^{(i)}) : i = 1, ..., m\right\}, \beta := 1, \mathbf{w}^{(1)} := \mathbf{0}, t := 1$
2: while $(t < T)$ do
3:     Pick randomly an example $(X^{(n)}, Y^{(n)})$ from $\mathcal{S}$
4:     $Y^{\star} = \arg\min_{Y \in \mathcal{Y}}(E(X, Y, \mathbf{w}) - \Delta(Y^{(n)}, Y))$
5:     $\mathcal{A}^n := \mathcal{A}^n \cup \{Y^{\star}\}$
6:     $\mathcal{A}^{n'} := \left\{Y \in \mathcal{A}^n : l(Y, Y^{(n)}, \mathbf{w}^{(t)}) > 0\right\}$
7:     $\eta^{(t)} := \frac{\beta}{t}$
8:     $\mathbf{g}^{(t)} := \frac{1}{\mathcal{A}^{n'}} \sum_{Y \in A^{n'}} \left(\Psi^D(Y^{(n)}) + \Psi^B(Y^{(n)}) - \left(\Psi^D(Y) + \Psi^B(Y)\right) + \frac{1}{C}\mathbf{w}\right)$
9:     $\mathbf{w}^{(t+1)} := \mathcal{P}\left[\mathbf{w}^{(t)} - \eta^{(t)}\mathbf{g}^{(t)}\right]$
10:    $t := t + 1$
11: end while
```

For performing loss-augmented inference, we employed the Fast-PD algorithm proposed by Komodakis et al. [1]. Fast-PD requires $B_{i,j}(\cdot, \cdot) \geq 0$.[4] The update of the weights (step 9) can potentially violate this constraint. Thus, we apply a projection $\mathcal{P}$ to ensure the compatibility of the weights $\mathbf{w}$ with Fast-PD.

3 Results

We evaluated our method via a 5-fold cross-validation on a subset of the BRATS2015 training data, encompassing 20 high-grade glioma cases (part of the former

[3] $l(Y^{(n)}, Y^{\star}, \mathbf{w}) = [E(X^{(n)}, Y^{(n)}, \mathbf{w}) + \Delta(Y^{(n)}, Y) - E(X^{(n)}, Y^{\star}, \mathbf{w})]_+$

[4] Fast-PD requires $B_{i,j}$ to define a semi-metric.

BRATS2013 training set). The performance of the presented method was compared against our previous approach using hand-tuned CRF parameters (baseline). Quantitative results are presented in table 1.

Region	Dice coefficient	Absolute volume error [mm^3]
Complete tumor (CRF+Learning)	$(0.887, 0.35)/(0.885, 0.35)$	$(10276, 41871)/(11078, 41257)$
Complete tumor (CRF Baseline)	$(0.888, 0.353)/(0.886, 0.353)$	$(9029, 42199)/(9029, 42001)$
Tumor core (CRF+Learning)	$(0.784, 0.912)/(0.793, 0.538)$	$(6504, 29505)/(6472, 29505)$
Tumor core (CRF Baseline)	$(0.789, 0.915)/(0.79, 0.58)$	$(6057, 32954)(6017, 32954)$
Enhancing tumor (CRF+Learning)	$(\mathbf{0.811}, 0.918)/(\mathbf{0.812}, 0.827)$	$(2784, 29875)/(2825, 29875)$
Enhancing tumor (CRF Baseline)	$(0.767, 0.942)/(0.768, 0.852)$	$(2485, 36986)/(2041, 36986)$

Table 1: Results of evaluation on subset of BRATS2015 training set. Performance measures are given as (median, range=max-min). Left tuple: Results for all 20 cases. Right tuple: Results after removal of outlier "brats_2013_pat0012_1 ".

4 Discussion and Future Work

The preliminary results indicate that learning CRF parameters from data instead of hand-tuning them can lead to comparable or even improved performance. Future work for our final submission will include training on the complete BRATS2015 training set and the investigation of more elaborate task-specific loss functions.

Acknowledgments. This project has received funding from the European Unions Seventh Framework Programme for research, technological development and demonstration under grant agreement N°600841.

References

1. Komodakis, N., Tziritas, G.: Approximate Labeling via Graph Cuts based on Linear Programming. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(8), 2007.
2. Lucchi, A., Marquez-Neila, P., Becker, C., Li, Y., Smith, K., Knott, G., Fua, P.: Learning Structured Models for Segmentation of 2D and 3D Imagery. IEEE Transactions on Medical Imaging (March), 2014.
3. Meier, R., Bauer, S., Slotboom, J., Wiest, R., Reyes, M.: Appearance- and Context-sensitive Features for Brain Tumor Segmentation. MICCAI BRATS Challenge Proceedings, 2014.
4. Menze, B.H., Jakab, A., et al.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). TMI 2014.
5. Ratliff, N.D., Bagnell, J.A., Zinkevich, M.A.: (Online) Subgradient Methods for Structured Prediction. Artificial Intelligence and Statistics, 2007.
6. Tsochantaridis, I., Hofmann, T., Joachims, T., Altun, Y.: Support Vector Machine Learning for Interdependent and Structured Output Spaces. ICML 2004.
7. Zhao, L., Wu, W., Corso, J.J.: Semi-Automatic Brain Tumor Segmentation by constrained MRFs using Structural Trajectories. MICCAI 2013.

Deep Convolutional Neural Networks for the Segmentation of Gliomas in Multi-Sequence MRI

Sérgio Pereira[1,2], Adriano Pinto[1], Víctor Alves[2], and Carlos A. Silva[1]

[1] MEMS-UMinho Research Unit, Guimarães, Portugal
id5692@alunos.uminho.pt, csilva@dei.uminho.pt
[2] Centro Algoritmi, Universidade do Minho, Braga, Portugal

Abstract. In their most aggressive form, gliomas are very deadly. Accurate segmentation is important for surgery and treatment planning, as well as for follow up evaluation. In this paper, we propose to segment brain tumors using a Deep Convolutional Neural Network (CNN). Neural Networks are known to suffer from overfitting. To address it, we use Dropout, leaky Rectifier Linear Units (ReLU), small convolutional kernels and small dense layers. We report preliminary, but promising results obtained using BraTS 2015 Training dataset.

Keywords: Magnetic Resonance Imaging (MRI), Brain Tumor, Glioma, Segmentation, Deep Learning, Deep Convolutional Neural Network

1 Introduction

Gliomas are a type of brain tumor that can be divided into Low Grade Gliomas (LGG) and High Grade Gliomas (HGG). Although the former are less aggressive, the later can be very deadly [2, 7]. In fact, the most aggressive gliomas are called Glioblastoma Multiforme, with most patients not surviving more than fourteen months, on average, even if they are under treatment [13]. The accurate segmentation of the tumor and its sub-regions is important for treatment and surgery planning, but also for follow-up evaluations [2, 7].

Over the years, several methods were proposed for brain tumor segmentation. In [2], Bauer et al. presents a broad survey on brain tumor image analysis methods in MRI. Some of the most successful methods employ supervised learning techniques, such as Random Forests [12] or Support Vector Machines [1].

All the previous methods require the computation of hand-crafted features, which may require specialized knowledge on the problem and may be difficult to design discriminative features. On the other hand, Deep Learning methods automatically extract features. In CNN, a set of filters are optimized and convolved with the input image to enhance certain characteristics. Those filters represent weights of the neural network. So, the same filters contribute to the same feature maps, making the weights shared across neural units. In this way, the number of parameters in these networks is lower than in neural networks of only fully connected layers, making them less prone to overfitting [5]. Another important mechanism against overfitting is Dropout [10]. Some methods employing CNN for brain tumor segmentation were already proposed [6, 4].

Inspired by Simonyan and Zisserman [9], we developed CNN architectures using only very small 3×3 kernels. In this way, we can have more convolutional layers, with the opportunity to apply more non-linear transformations. We report preliminary results using BraTS 2015 Training dataset. Although we present here the best performing architecture for each grade, they will be subjected to some small improvements for the final contest.

2 Materials and Methods

In this preliminary implementation, the CNN takes as input a patch extracted in the axial plane of all the available MRI sequences. The processing pipeline has three main stages: pre-processing, classification and post-processing. Given the differences between HGG and LGG, it was trained a model for each grade.

2.1 Data

The Training dataset of BraTS 2015 comprises 220 acquisitions from patients with HGG and 54 from patients with LGG. For each patient there are available four MRI sequences: T1-, contrast enhanced T1- (T1c), T2- and T2-weighted FLAIR. All images were already aligned with the T1c and skull stripped.

2.2 Method

Pre-processing All images were inhomogeneity corrected using the N4ITK method [11]. After that, the histogram of each individual sequence was normalized [8]. Finally, each sequence was transformed to have zero mean and unit standard deviation.

Convolutional Neural Network The architectures of the CNNs were developed following [9], being described in Table 1. HGG allowed a deeper architecture than LGG. The input consists in 33×33 axial patches in each of the 4 MRI sequences. Max-pooling is performed with some overlapping of the receptive fields. In all the fully-connected layers we use Dropout with $p = 0.5$. The loss function was Categorical Cross-entropy, optimized through Stochastic Gradient Descent with Nesterov‘s Momentum. The CNN was implemented using Theano [3].

Post-processing A morphological filter was applied to remove isolated clusters.

3 Results and Discussion

Preliminary results were obtained using BraTS 2015 Training Dataset, as presented in Table 2 and Figure 1. In HGG it was used 2-fold cross-validation, while for LGG the results were obtained with 3-fold cross-validation. Results on LGG are lower than in HGG. This may be due to the lower contrast and smaller size of the LGG. Additionally, there are less available training cases in LGG than in HGG. The entire processing pipeline takes less than 10 minutes to segment each patient, using GPU processing with a Nvidia Geforce GTX 980.

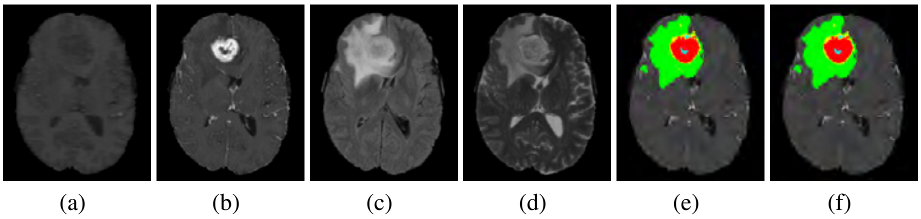
Table 1. Architecture of the CNN for HGG (left) and LGG (right). The number following *r* corresponds to the receptive field and *s* to the stride. After ”-” it is indicated the number of sequences in the input, the number of filters in convolutional layers or the number of nodes in the fully-connected layers. All non-linearities were Leaky ReLU.

Input (33 × 33 × 4)
Conv. r3 s1 - 64
Conv. r3 s1 - 64
Conv. r3 s1 - 64
Max-pooling r3 s1
Conv. r3 s1 - 128
Conv. r3 s1 - 128
Conv. r3 s1 - 128
Max-pooling r3 s1
Fully-connected - 256
Fully-connected - 256
Fully-connected (soft-max) - 5

Input (33 × 33 × 4)
Conv. r3 s1 - 64
Conv. r3 s1 - 64
Max-pooling r3 s1
Conv. r3 s1 - 128
Conv. r3 s1 - 128
Max-pooling r3 s1
Fully-connected - 256
Fully-connected - 256
Fully-connected (soft-max) - 5

Table 2. Results obtained using BraTS 2015 Training dataset.

	Dice			Positive Predictive Value			Sensitivity		
	Complete	Core	Enhanced	Complete	Core	Enhanced	Complete	Core	Enhanced
LGG	0.86	0.64	0.40	0.86	0.67	0.39	0.88	0.71	0.51
HGG	0.87	0.75	0.75	0.89	0.76	0.80	0.86	0.79	0.75
LGG + HGG	0.87	0.73	0.68	0.89	0.74	0.72	0.86	0.77	0.70



(a) (b) (c) (d) (e) (f)

Fig. 1. Subject 199 from the BraTS 2015 HGG Training dataset. a) T1. b) T1c. c) Flair. d) T2. e) Manual segmentation. f) Automatic segmentation. Blue - necrosis, green - edema, yellow - non-enhanced tumor, red - enhanced tumor.

4 Conclusions and Future Work

In this work, we implemented a CNN to segment brain tumors in MRI. All the processing pipeline is fully automatic. Although simple, this architecture shows promising results. We used just axial patches. So, in the future, we intend to extend the method to include information from the remaining planes. Until the challenge, the architecture and parameters may be slightly changed.

Acknowledgments Sérgio Pereira was supported by a scholarship from the Fundação para a Ciência e Tecnologia (FCT), Portugal (scholarship number PD/BD/105803/2014).

References

1. Bauer, S., Nolte, L.P., Reyes, M.: Fully automatic segmentation of brain tumor images using support vector machine classification in combination with hierarchical conditional random field regularization. In: Fichtinger, G., Martel, A., Peters, T. (eds.) MICCAI 2011, LNCS, vol. 6893, pp. 354–361. Springer Berlin Heidelberg (2011)
2. Bauer, S., Wiest, R., Nolte, L.P., Reyes, M.: A survey of mri-based medical image analysis for brain tumor studies. Phys. Med. Biol. 58(13), R97 (2013)
3. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y.: Theano: a CPU and GPU math expression compiler. In: Proceedings of the Python for Scientific Computing Conference (SciPy) (Jun 2010)
4. Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.M., Larochelle, H.: Brain tumor segmentation with deep neural networks. arXiv preprint arXiv:1505.03540 (2015)
5. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature 521(7553), 436–444 (2015)
6. Lyksborg, M., Puonti, O., Agn, M., Larsen, R.: An ensemble of 2d convolutional neural networks for tumor segmentation. In: Image Analysis, pp. 201–211. Springer (2015)
7. Menze, B., et al.: The multimodal brain tumorimage segmentation benchmark (brats). IEEE Trans. Med. Imaging (2014)
8. Nyúl, L.G., Udupa, J.K., Zhang, X.: New variants of a method of mri scale standardization. IEEE Trans. Med. Imaging 19(2), 143–150 (2000)
9. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
10. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research 15(1), 1929–1958 (2014)
11. Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C.: N4itk: improved n3 bias correction. IEEE Trans. Med. Imaging 29(6), 1310–1320 (2010)
12. Tustison, N.J., Shrinidhi, K., Wintermark, M., Durst, C.R., Kandel, B.M., Gee, J.C., Grossman, M.C., Avants, B.B.: Optimal symmetric multimodal templates and concatenated random forests for supervised brain tumor segmentation (simplified) with antsr. Neuroinformatics pp. 1–17 (2014)
13. Van Meir, E.G., Hadjipanayis, C.G., Norden, A.D., Shu, H.K., Wen, P.Y., Olson, J.J.: Exciting new advances in neuro-oncology: The avenue to a cure for malignant glioma. CA: a cancer journal for clinicians 60(3), 166–193 (2010)

Brain Tumor Segmentation with Deep Learning

Vinay Rao , Mona Sharifi Sarabi , Ayush Jaiswal

University of Southern California

Abstract. Deep Neural Networks (DNNs) have recently shown outstanding performance on image classification and segmentation tasks. This paper presents our work on applying DNNs to brain tumor segmentation for the BRATS 2015 challenge. Our approach to finding tumors in brain images is to perform a pixel-wise classification. We learn deep representations for each pixel based on its neighborhood in each modality (T1, T1c, T2 and Flair) and combine these to form a multimodal representation for each pixel. We present preliminary results of our work in this paper. We also outline our future steps and experiments, which involve learning joint multimodal representations of the pixels based on recent work published in Deep Learning literature.

1 Introduction

Segmenting brain tumors in multi-modal imaging data is a challenging problem due to unpredictable shapes and sizes of tumors. Deep Neural Networks (DNNs) have already been applied to segmentation problems and have shown significant performance improvement compared to the previous methods [4]. We use Convolutional Neural Networks (CNNs) to perform the brain tumor segmentation task on the large dataset of brain tumor MR scans provided by BRATS2015.

CNNs are DNNs in which trainable filters and local neighborhood pooling operations are applied alternatingly on the raw input images, resulting in a hierarchy of increasingly complex features. Specifically, we used multi-modality information from T1, T1c, T2 and Flair images as inputs to different CNNs. The multiple intermediate layers apply convolution, pooling, normalization, and other operations to capture the highly nonlinear mappings between inputs and outputs. We take the output of the last hidden layer of each CNN as the representation of a pixel in that modality and concatenate the representations of all the modalities as features to train a random forest classifier.

2 Data Analysis

The BRATS dataset consists of both high-grade and low-grade gliomas with four modalities: T1, T1c, T2 and Flair. We visualized the two types of gliomas and found clear visual differences, and hence, we treat finding tumors in them as separate problems. We used BrainSuite[1] to run a naïve histogram classification algorithm on the dataset to extract Cerebral Spinal Fluid(CSF) patches, along with gray and white matter data by using this software. Fig 1 shows the output of the software for one of the samples. CSF is usually found outside the brain, but when it is found inside the brain, it is an indicator of the presence of abnormality.

3 Methods

Our approach to finding tumors in brain images is to perform pixel-wise classification. We extract 32x32 patches in XY, YZ and XZ planes around each pixel for each modality. We use a Deep Convolutional Neural Network (CNN) for each modality to learn good representations for every pixel based on the patches extracted surrounding that pixel. Each CNN is trained separately to classify a pixel as one of non-tumor, necrosis, edema, non-enhancing, and enhancing.

Each of the CNNs follows the architecture as in Fig 2. Raw pixels from patches around each pixel form the input to the network. The softmax layer classifies the pixel as one of the five classes. We use a rectified linear unit (ReLU) in conjunction with the final hidden layer to improve gradients.

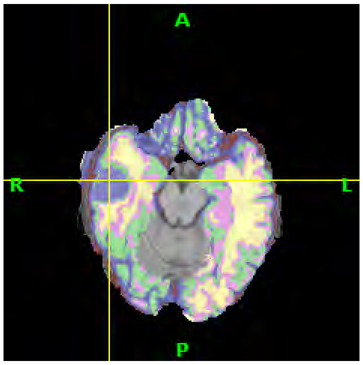


Fig. 1. CSF visualization through histogram classification

We performed experiments under two settings. In the first setting, we sample a random population of patches with equiprobable frequencies. The second setting makes use of all the patches from 20 randomly selected patients for training, and 5 for testing.

We take the output of the last hidden layer of each CNN as the representation of a pixel in that modality. We use the concatenation of the representations of all the modalities as features to train a random forest classifier. Fig 3 shows the transformation from raw pixels to final representations which are then classified by the random forest classifier.

4 Implementation

For training and definition of the CNNs, we make use of Caffe [2]. We use the ITK [3] library to prepare inputs for training the networks. We train the network using Stochastic Gradient Descent. We use the final representations learned by the four CNNs to train a random forest classifier using scikit-learn [4].

5 Results

In the first setting we trained the network with patches around 25000 randomly chosen pixels. We sampled the pixels so that their labels were inline to the distribution of the labels in the entire dataset. We were able to achieve an accuracy of 67% on a similarly sampled testing dataset. In the second setting we trained the network using all the patches of 10 patients and were able to reach a loss of 2.9 % on the training set. We are currently in the process of training and testing the network on bigger datasets using our high performance computing resources. All the preliminary results were run on workstations with 16GB of RAM and a CUDA compatible Nvidia GPU.

6 Future Work

We consider many areas for our ongoing and future work on this problem. In the experiments that we have run so far and reported in this paper, we have learned deep representations for each modality separately and concatenated them to form a single combined representation. We consider learning a joint representation from all the modalities together as a next step. There are multiple ways to do this. One way is to have a single CNN that takes all the modalities as input and learns a deep representation from the combined input. We will also try some methods developed very recently in multimodal deep representation learning research [1–3]. This is particularly applicable with this dataset because we can clearly treat T1, T1c, T2

[1] http://brainsuite.org/
[2] http://caffe.berkeleyvision.org/
[3] http://www.itk.org/
[4] http://scikit-learn.org

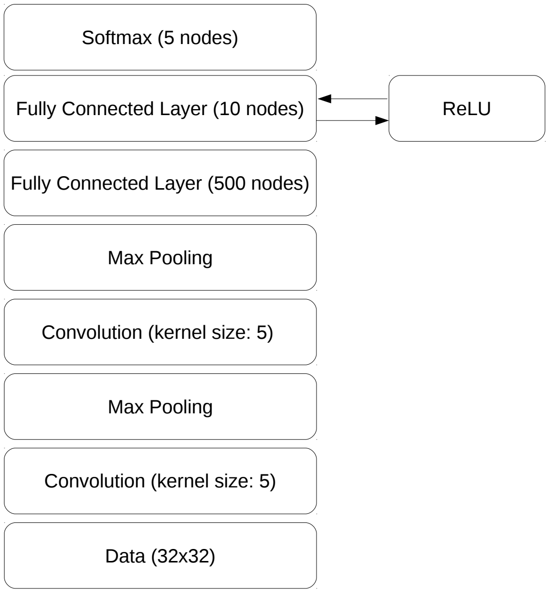

Softmax (5 nodes)
Fully Connected Layer (10 nodes)
ReLU
Fully Connected Layer (500 nodes)
Max Pooling
Convolution (kernel size: 5)
Max Pooling
Convolution (kernel size: 5)
Data (32x32)

Fig. 2. Architecture of each CNN

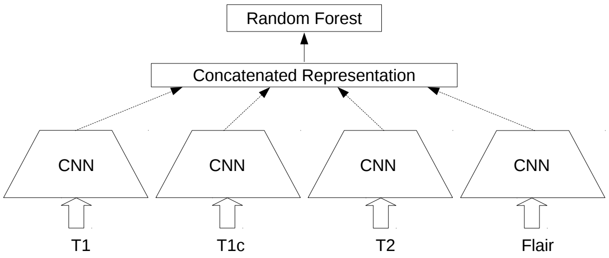


Fig. 3. Stacked Prediction Framework

and Flair as different modalities describing the same data or objects. These methods have been reported to learn better representations of mutlimodal data as compared to having a single network that takes all the modalities as combined input.

With respect to better learning in the deep neural networks, we plan to incorporate average and fractional pooling instead of max pooling. With respect to better learning in the deep neural networks, we plan to incorporate stochastic [6] and fractional max-pooling [5] as they have shown to improve the overall performance of models.

Another direction in which we plan to proceed in our future work is to incorporate the observation that the label of a pixel is influenced by the labels of surrounding pixels. We consider an ensemble method of classifiying all the pixels in a patch at once instead of classifying only the center pixel at a time. We then pick the final label for each pixel by voting. On similar lines of using ensemble models, we will also work on an ensemble of classifiers that are learned on patches of different sizes (or zoom levels).

We also plan to incorporate expert features into our model based on the histograms generated by the BrainSuite software that we referred to earlier. We expect our models to perform better with these augmented features as they add highly informative complex information to the data.

Yet another area of future work is to choose a good training set that can give the classifiers good examples of difficult cases such as borders and non-empty non-tumorous patches. We will also look into trying small

3D regions around each pixel to label it as compared to using 2D patches as we are currently doing. Along these lines, we also plan to look at some heuristic ways to provide good intial weights to pixels that might provide more information towards the classification of the center pixel based on density-based clustering techniques.

References

1. W. Wang, R. Arora, K. Livescu, J. Bilmes, On Deep Multi-View Representation Learning, Proceedings of The 32nd International Conference on Machine Learning, pp. 1083–1092, 2015
2. K. Sohn, W. Shang, H. Lee, Improved Multimodal Deep Learning with Variation of Information, Advances in Neural Information Processing Systems 27, pp. 2141–2149, 2014
3. N. Srivastava, R. Salakhutdinov, Multimodal Learning with Deep Boltzmann Machines, Journal of Machine Learning Research, vol. 15, pp. 2949–2980, 2014
4. W. Zhang, R. Li, H. Deng, L. Wang, W. Lin, S. Ji, D. Shen, Deep convolutional neural networks for multi-modality isointense infant brain image segmentation, NeuroImage, vol. 108, pp. 214–224, 2015
5. Benjamin Graham, Fractional-Max Pooling, International Conference on Learning Representations, 2015
6. Matthew D. Zeiler, Rob Fergus, Stochastic Pooling for Regularization of Deep Convolutional Neural Networks,

Multi-Modal Brain Tumor Segmentation Using Stacked Denoising Autoencoders

Kiran Vaidhya[*], Roshan Santhosh[*], Subramaniam Thirunavukkarasu[*], Varghese Alex[*], and Ganapathy Krishnamurthi[*]

Indian Institute of Technology Madras, Chennai
gankrish@iitm.ac.in

Abstract. Automatic segmentation of Gliomas from Magnetic Resonance Images(MRI) is of great importance as manual segmentation is time consuming and the inter-rater variability is high. Autoencoders have been shown to learn good features for classification in natural images and digits dataset. In this paper, we make use of Autoencoders in medical imaging to automate segmentation of Gliomas from MRI. A 3 layer over-complete Stacked Denoising Autoencoder(SDAE), trained with a combination of unsupervised and supervised learning techniques, was used for this task. From our experiments, we achieved a preliminary dice score of 81.41% for whole tumor segmentation and there is still scope for improvement.

Keywords: Gliomas, MRI, SDAE, Unsupervised Learning, Supervised Learning

1 Introduction

Autoencoders are fully connected neural networks that are trained to reconstruct the given input. The concept of unsupervised "pre-training" that makes the network learn a good representation of the data and supervised "fine-tuning" for classification revolutionized the area of deep learning [7].

SDAE, a variant of regular Autoencoder, is trained to reconstruct the original data from corrupted data [12]. By doing so, the SDAE learns to produce useful higher level representation of the input data. The input is noised either by Gaussian, Masking or Salt and pepper noise. SDAE has shown promising results in digit recognition and natural image classification tasks[4],[13]. The use of SDAE for classification task has been very limited in the medical domain. SDAE have been used for organ detection [10] and for characterizing the skin from OCT images[9]. A variant of Autoencoder has been used for detecting various stages of dementia[8].

[*] All authors have contributed equally

2 Materials and Methods

2.1 Pre-Processing

Dataset from BRATS 2015 was taken and minimally pre-processed in a similar way as explained in [11] using histogram matching and removing outliers.

Patches from the images were extracted and fed to the network. However, we encountered class imbalance issues as most of the patches extracted corresponded to normal or healthy tissues while the number of patches centered around a lesion pixel were relatively lower. The severity of the class imbalance was reduced by extracting patches only from in and around the tumor.

2.2 Details and Architecture of Network used

The network was trained with 21 patients, validated on 10 patients and tested on 24 patients. 3D patches were extracted from all four sequences and concatenated to form the input layer of the SDAE. The size of the patch was 9x9x9 with a fixed overlap between subsequent patches. Various other sizes for patch extraction like 7x7x7, 5x5x5, 3x3x3 and 2D patch sizes like 11x11, 9x9, 21x21, 15x15, 13x13 were experimented for a range of over-complete and under-complete architectures with varying levels of masking noise.

We observed that the hidden layer architecture of 5000-2000-500 with 5 class outputs and masking noise of 10% in each layer gave us the best results. Pre-training was carried with equal number of patches from each class while, in fine-tuning, the class balance was proportional to that of the original image. Dynamic learning rates and penalty for specific classes in the fine-tuning cost were implemented. The network was pre-trained using RmsProp [6] while the fine-tuning was optimised using Stochastic Gradient Descent. In the pre-training, sigmoid activation function was used for encoding while linear activation function was used for decoding. For fine-tuning, soft-max activation function was used for the final layer.

Ratio of patches from Normal:Necrotic:Edema:Non-Enhancing:Enhancing regions was in the order of 81:1:12:2:4.

2.3 Post Processing

Post processing, comprising of connected component analysis and applying cerebellum and brain-stem masks, was done to eliminate false positives. The masks were obtained through Atropos segmentation [3]

3 Results and Discussion

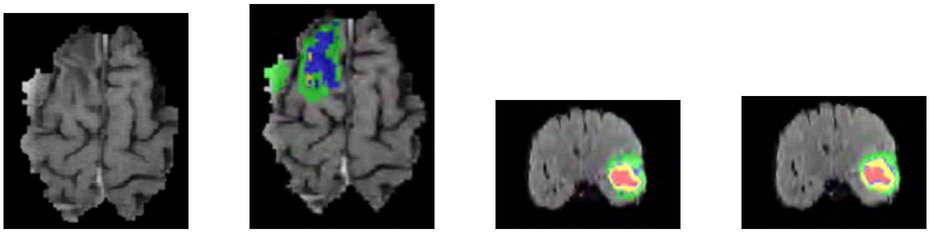
3.1 Results

We report our best performing network on the test images table 1, which has a mean whole tumor dice score of $81.41\% \pm 9.6\%$, mean active tumor dice score of

50.97% ± 29.33%and a mean tumor core dice score of 60.63% ± 25.7%. The dice scores were calculated using Advanced Normalisation Toolkit software[1].

For whole tumor, sensitivity is 80.89% and specificity is 84.51%. For tumor core, sensitivity is 67.54% and specificity is 60.22%. For active tumor, sensitivity is 82.61% and specificity is 42.95%.

For tumor core and active tumor classification tasks, the algorithm performed below the expected performance for certain patients, for example, patient ID 374 shown in Fig. 1 (a-b). A possible explanation for such behaviour would be that, the amount of pixels corresponding to enhancing tumor were very low, hence, missing out on them would have a huge impact on the mean active and tumor core dice scores. Excluding patient 374 from the test dataset, the mean whole tumor dice score was found to be 83.79% ± 9.7% , mean active tumor dice score of 66.05% ± 17.7% and the mean tumor core dice score of 72.4% ± 16.6%.



(a) T1c

(b) Ground Truth overlaid on T1c

(c) Ground Truth overlaid on FLAIR

(d) Prediction overlaid on FLAIR

Fig. 1: (a) and (b) - Worst Performing image, as amount of enhancing tumor is low. (c) and (d) - Best performing image. For all Images, Green - Edema, Blue - Non Enhancing Tumor, Yellow - Enhancing Tumor, Red - Necrotic core

3.2 Discussion

As stated in [5], we found data imbalance to be the major issue as the ratio of necrotic core and non-enhancing tumor voxels was lower than that of edema. We implemented a penalty in the cost function for the respective classes and found the mean dice scores to improve. However there were a few patients where the dice scores have dropped and we are currently experimenting on this. Data augmentation and duplication can be explored to get better results. Our programs were written on Python using Theano package[2] and were run on K20 and GTX-980 GPUs.

4 Conclusion

In this paper, we present a fully automatic method to segment brain tumor using Stacked Denoising Autoencoder. The algorithm achieves a mean whole tumor dice score of 81.41% on the test data with a standard deviation of 9.6%, which is comparable to the top scores reported in BRATS 2014 and the standard deviations are comparable to the inter-rater variability in manual segmentation. There is still scope for improvement by implementing dropouts, sparsity and deeper architectures.

Table 1: Performance on Test Data

Patient Id	Whole Tumor			Tumor Core			Active Tumor		
	Dice	FN	FP	Dice	FN	FP	Dice	FN	FP
pat105-0001	.92	.10	.037	.87	.14	.105	0.82	.08	.26
pat111-0001	.87	.10	.14	.71	.07	.42	.56	.057	.59
pat113-0001	.86	.22	.022	.77	.26	.19	.74	.18	.30
pat117-0001	.83	.25	.054	.68	.45	.064	.76	.25	.20
pat118-0001	.93	.07	.043	.90	.072	.114	.84	.03	.24
pat120-0001	.85	.07	.201	.77	.11	.310	.74	.08	.36
pat192-0001	.93	.10	.032	.90	.03	.15	.90	.03	.14
pat193-0002	.88	.07	.139	.23	.06	.86	.24	.007	.85
pat198-0001	.57	.046	.20	.40	.26	.71	.30	.05	.81
pat198-0283	.83	.26	.0591	.67	.35	.28	.79	.07	.30
pat226-0001	.85	.17	.074	.80	.24	.13	.79	.06	.31
pat226-0090	.87	.24	.073	.81	.25	.10	.69	.07	.44
pat309-0001	.77	.26	.172	.68	.40	.18	.51	.42	.53
pat309-0120	.91	.10	.068	.72	.04	.41	.53	0.01	.63
pat374-0557	.69	.34	.256	.38	.60	.618	.11	.27	.93
pat374-0801	.78	.28	.140	.23	.79	.71	.03	.21	.98
pat374-0909	.77	.27	.172	.25	.77	.702	.04	.44	.97
pat374-1165	.68	.41	.165	.14	.87	.82	.008	.81	.99
pat374-1426	.73	.38	.106	.10	.93	.666	.04	.63	.97
pat374-1627	.78	.24	.174	.38	.51	.68	.09	.11	.95
pat375-0001	.90	.09	.100	.86	.10	.170	.72	.17	.34
pat399-0595	.79	.25	.143	.75	.17	.305	.60	.05	.55
pat498-0001	.80	.15	.22	.85	.21	.130	.74	.21	.30
pat499-0001	.62	.06	.532	.62	.06	.531	.53	.06	.62

References

[1] B. Avants, N. Tustison, and G. Song. Advanced normalization tools: V1.0. 07 2009.

[2] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. J. Goodfellow, A. Bergeron, N. Bouchard, and Y. Bengio. Theano: new features and speed improvements. 2012.

[3] C. Durst, N. Tustison, M. Wintermark, and B. Avants. Ants and arboles. 2013.

[4] X. Glorot, A. Bordes, and Y. Bengio. Deep sparse rectifier neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.

[5] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. C. Courville, Y. Bengio, C. Pal, P. Jodoin, and H. Larochelle. Brain tumor segmentation with deep neural networks. *CoRR*, abs/1505.03540, 2015. URL http://arxiv.org/abs/1505.03540.

[6] G. Hinton, N. Srivastava, and K. Swersky. Neural networks for machine learning lecture 6e rmsprop :divide the gradient by a running average of its recent magnitude.

[7] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[8] S. Liu, S. Liu, W. Cai, S. Pujol, R. Kikinis, and D. Feng. Early diagnosis of alzheimer's disease with deep learning. In *Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on*, pages 1015–1018, April 2014. doi: 10.1109/ISBI.2014.6868045.

[9] D. Sheet, S. P. K. Karri, A. Katouzian, N. Navab, A. K. Ray, and J. Chatterjee. Deep learning of tissue specific speckle representations in optical coherence tomography and deeper exploration for in situ histology. pages 777–780, 2015.

[10] H.-C. Shin, M. Orton, D. Collins, S. Doran, and M. Leach. Stacked autoencoders for unsupervised feature learning and multiple organ detection in a pilot study using 4d patient data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1930–1943, Aug 2013. ISSN 0162-8828. doi: 10.1109/TPAMI.2012.277.

[11] S. Vaidya, A. Chunduru, R. Muthuganapathy, and G. Krishnamurthi. Longitudinal multiple sclerosis lesion segmentation using 3d convolutional neural networks.

[12] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research*, 11:3371–3408, 2010.

[13] N. Wang and D.-Y. Yeung. Learning a deep compact image representation for visual tracking. In *Advances in Neural Information Processing Systems*, pages 809–817, 2013.