Local Structure 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 re-

quires 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.

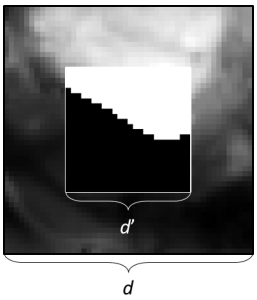


Fig. 1. Local structure prediction: Image feature patches (with side length d) are used to predict the most likely label patch (with side length d') in its center. While standard patch based prediction approaches use $d' = 1$ (voxel), we consider in this paper all values with $1 \leq d' \leq d$.

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.

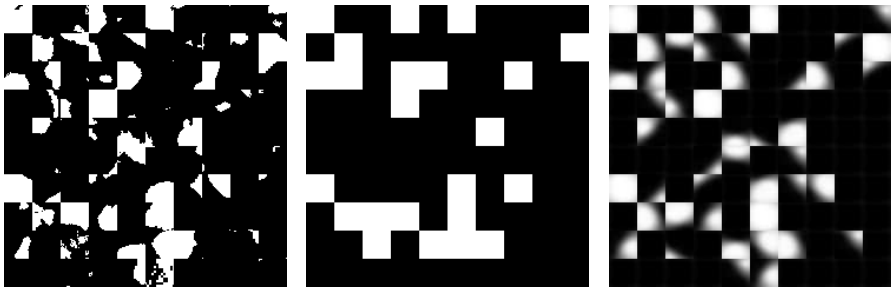
Label patches. Treating the annotation task for each class individually, we obtain a label space $\mathcal{L} = \{0, 1\}$ that is given by an expert's manual segmentation of the pathological structures. The *label patch* is then a patch $\mathbf{p}$ of size $d' \times d'$ from the structured label space $\mathcal{P}$, 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.



(a) (b) (c)

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.

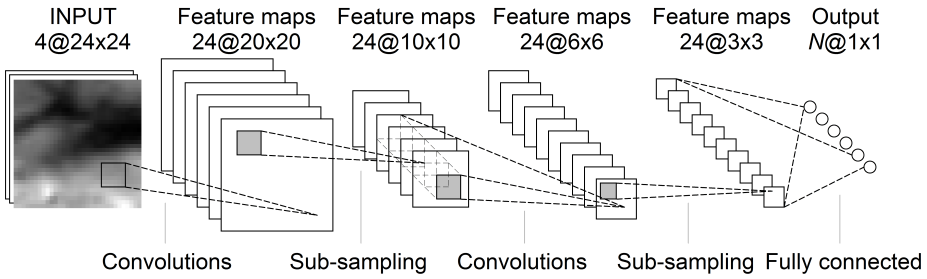


Fig. 3. Architecture of Convolutional Neural Network for $d = 24$. The input of the network is a multimodal image patch. The output of the network are N probabilities, where N denotes the size of label patch dictionary.

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

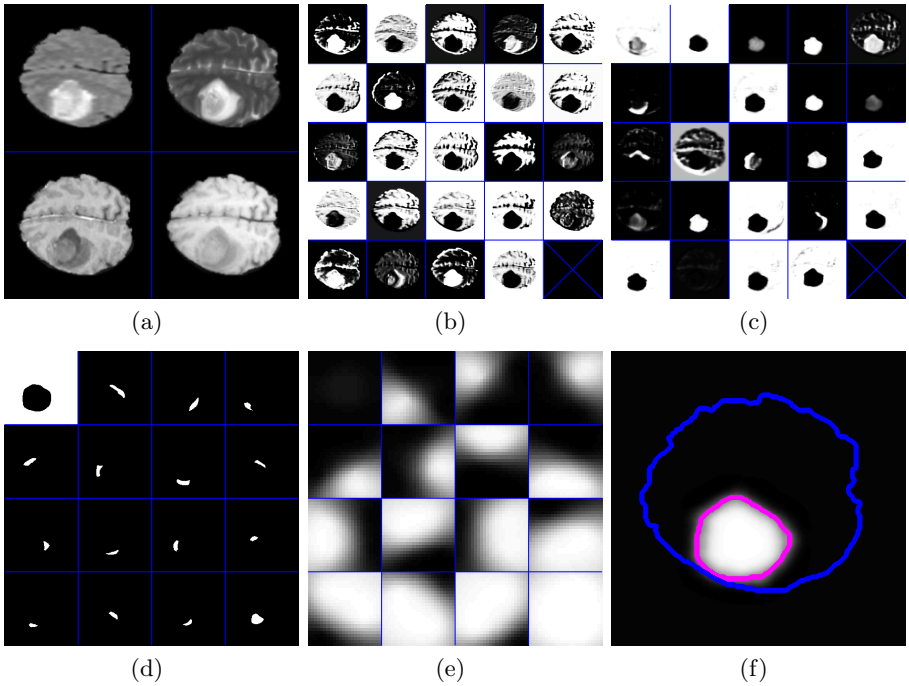


Fig. 4. Sequential processing of multimodal slice (a). (b) and (c) show all 24 outputs of the first and the second convolutional layer. (d) depicts the output of the whole CNN architecture for given 16 groups with average patch labels depicted in (e). (f) shows the final probability map of the whole 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 $240\times240\times155$. 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/

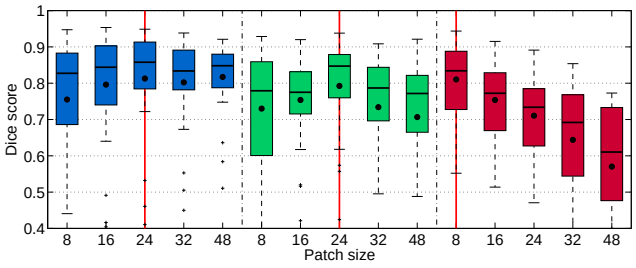


Fig. 5. Dice score as a function of the **image patch size** d with its best label patch size d' with label patch dictionary size $N = 32$ for the whole tumor (blue), tumor core (green) and active tumor(red).

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.

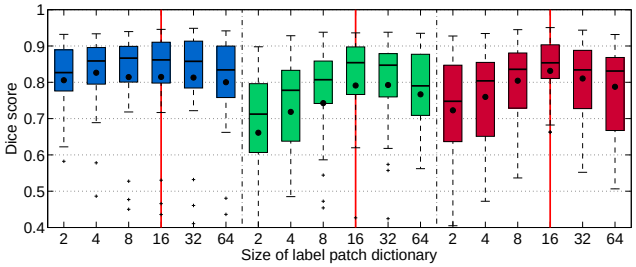


Fig. 6. Dice score as a function of **label patch dictionary size** N using the optima of Fig. 5: $d = 24$ for whole tumor (blue), $d = 24$ for tumor core (green), $d = 8$ for active tumor (red).

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.

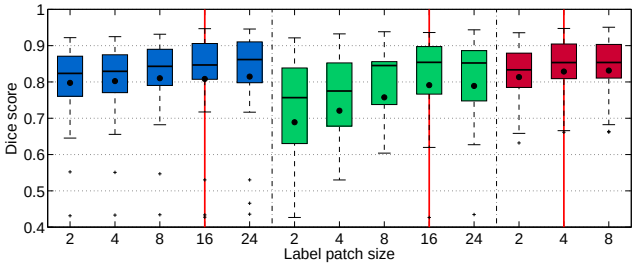


Fig. 7. Dice score as a function of **label patch size** d' for whole tumor (blue) with $d = 24$, tumor core (green) with $d = 24$, and active tumor (red) with $d = 8$, with label patch dictionary size $N = 16$.

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 resolution	Slice step	Computing time per volume	Dice Score (in%) 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.

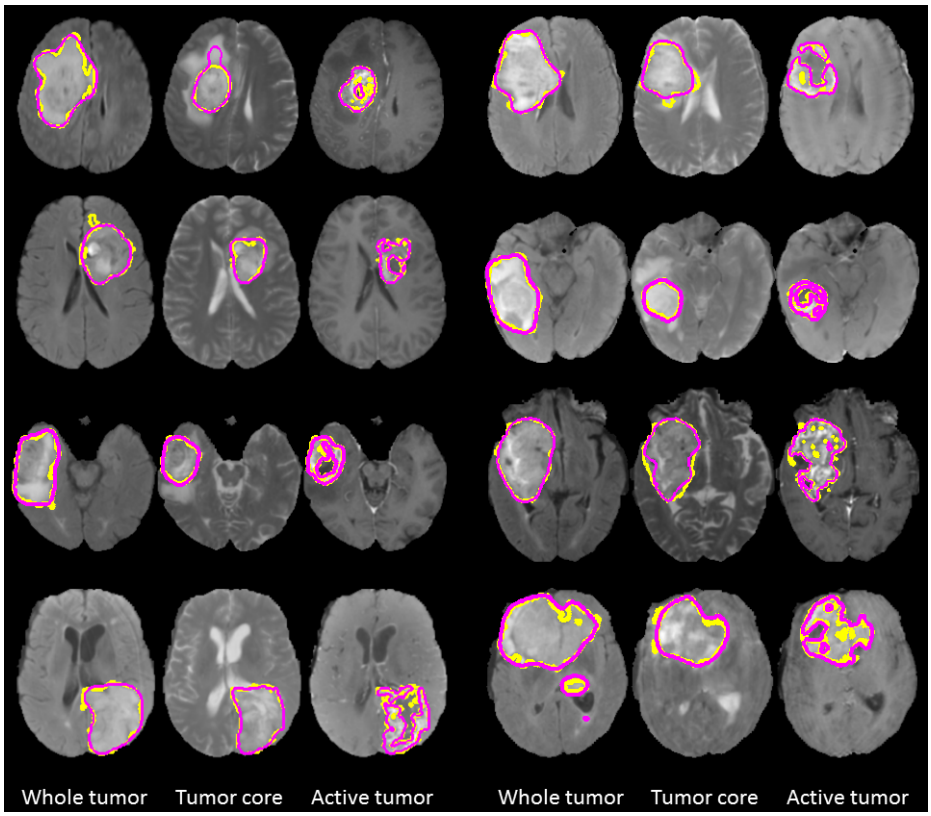


Fig. 8. Example of consensus expert annotation (yellow) and automatic segmentation (magenta) applied to the test image data set. Each row shows two cases. From left to right: segmentation of whole tumor (shown in FLAIR), tumor core (shown in T2) and active tumor (shown in T1c).

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)