LIVER TUMOR SEGMENTATION CT DATA BASED ON ALEXNET-LIKE CONVOLUTION NEURAL NETS

A.N. Korabelnikov[1], A.V. Kolsanov[2], S.S. Chaplygin[2], P.M. Zelter[2],
K.V. Bychenkov[3], A.V. Nikonorov[1]

[1]Samara National Research University, Samara, Russia
[2]Samara State Medical University, Samara, Russia
[3]"MEDEKS", LLC, Samara, Russia

Abstract. Anatomical structure segmentation on computed tomography (CT) is the key stage in medical visualization and computer diagnosis. Tumors are one of types of internal structures, for which the problem of automatic segmentation today has no solution fully satisfying by quality. The reason is high variance of tumor's density and inability of using a priori anatomical information about shape. In this paper we propose automatic method of liver tumors segmentation based on convolution neural nets (CNN). Studying and validation have been performed on set of CT with liver and tumors segmentation ground truth. Average error (VOE) by cross-validation is 17.3%. Also there were considered algorithms of pre- and post-processing which increase accuracy and performance of segmentation procedure. Particularly the acceleration of the segmentation procedure with negligible decrease of quality has been reached 6 times.

Keywords: deep learning, convolutional neural networks, computed tomography, segmentation, liver tumors, computer-aided diagnosis, computer-aided surgery, CADx, CADe, medical imaging, radiology

Citiation: Korabelnikov AN, Kolsanov AV, Bychenkov KV, Nikonorov AV. Liver tumor segmentation CT data based on AlexNet-like convolution neural nets. CEUR Workshop Proceedings, 2016; 1638: 348-356. DOI: 10.18287/1613-0073-2015-1638-348-356

1 Introduction

Automatic segmentation of DICOM data, obtained from CT or MRI is an important prerequisite for computer-aided diagnosis (CADx) and computer-aided detection (CADe). Liver tumors segmentation is necessary for planning treatment and choosing treatment options such as tumor resection, image-guided radiation therapy (IGRT), radiofrequency ablation etc. The purpose of this paperwork is development and analysis of the automatic segmentation method based on convolutional neural networks (CNN).

Image segmentation is a labeling each pixel of the image by common visual characteristics [6]. In this work, the image has divided into two segments: tumors and healthy liver tissues. In works [1-4] segmentation was performed by different methods: the watershed method, threshold methods, the graph cut method and classification with features selection. In work [5] deep learning methods for pancreas segmentation was applied.

Opposite to organ's segmentation problem the liver tumor's segmentation problem is complicated by a large diversity of location, shape and density which varies from one patient to another. These factors don't let to use shape as a priori information that significantly increases quality in other cases. [7].

Different metrics are used as segmentation's quality criteria so there is no unified or common metric standard [8, 9]. In this works VOE (Volume Overlapping Error) has been used as quality criteria.

In this work segmentation method based on CNN has been developed. Also methods of pre- and post-processing which are aided to quality and performance increase have been considered. Furthermore additional dividing tumor's class on subclasses can significantly increase quality in contrast with ordinal two-class division (healthy/affected tissue).

2 Segmentation as classification through CNN

Segmentation is made by assigning each voxel of image to one of two classes (healthy or affected tissue). Input of classifier is a 2D neighborhood of voxel (volume pixel) on native plane. Output is a class label of this voxel. In this approach, CNN has been used as a classifier. The classifier processes all voxels of liver's region. Liver segmentation is supposed to be previously constructed.

Convolutional neural networks (CNN), considered in the paper, [10] are evolution of multilayer artificial neural networks with backpropagation training. CNN layers consist of neurons with nonlinear activation function and learned weights. Network training is a process of searching weights values, which minimize classification error on training data. It is important to notice, all traits of ordinary nets are still applicable for CNNs.

CNN can be conventionally divided into two parts (convolutional and fully-connected (FC)). The first part is responsible for feature extraction, the second one implements classifier based on extracted features. In such a way CNN allow to solve optimal feature representation problem as well as classification for current training data.

Neural networks take a vector of value on their input and transform it through series of hidden layers. Each hidden layer is built of a bunch of neurons, where each neuron is connected to each other from previous layer. Moreover neurons are independent and do not share their weights. The last FC layer in CNN is called output layer. In case of classification, it represents the probabilities of classes [11].

2.1 Used CNN architecture

A network called AlexNet was proposed for semantic segmentation [12]. In this paper we have modified this network to take grayscaled images as the input and to return 2 (or 3) classes as the output.

Table 1. CNN's architecture based on AlexNet.

№	Type (kernel size, stride, padding)	Output size	The depth (number of outputs)
0	Input data	64x64	1
1	Convolution (11,4,0)	14x14	64
2	ReLU	14x14	64
3	LRN	14x14	64
4	Pooling (3,2)	7x7	64
5	Convolution (5,1,2)	7x7	192
6	ReLU	7x7	192
7	LRN	7x7	192
8	Pooling (3,2)	3x3	192
9	Convolution (3,1,1)	3x3	384
10	ReLU	3x3	384
11	Convolution (3,1,1)	3x3	256
12	ReLU	3x3	256
13	Convolution (3,1,1)	3x3	256
14	ReLU	3x3	256
15	Pooling (3,2)	1x1	256
16	Fully Connected	1x1	4096
17	Dropout, rate 0.5	1x1	4096
18	Fully Connected	1x1	4096
19	ReLU	1x1	4096
20	Dropout, rate 0.5	1x1	4096
21	Fully Connected	1x1	2 (3)
22	SoftMax	1x1	2 (3)

The table 1 presents multilayer structure of the used net. The specific of convolution and pooling layers is decreasing of spatial dimensions and increasing the depth. The depth (last column) is a number of outputs with different coefficients. The convolution layer performs a the number of convolutions between windows (its receipted filed and its weights). The pooling layer performs an operation of maximum on some windows (its receipted field). The standard layer ReLU (rectified linear unit) performs piecewise linear transformation max(x, 0). The LRN (local response normalization) layer performs norming by input data (by default in depth domain). The FC layer performs scalar product, taking a vector at the input and returning a scalar. The dropout layer drops a part of neurons on training stage thereby preventing the network from overfitting [11, 12].

2.2 CNN implementation

As the software platform we chose the popular Caffe framework. The key points which makes the Caffe really useful for our research:

- declarative style of programming CNN architecture and training process;
- repository of models deigned for Caffe framework has become de-facto standard in deep learning [13];
- flexibility of execution on CPU as well as on GPU and wide support of this framework by Nvidia cuDNN [8];
- open sources C++ implementation and community support;
- wrappers for Python and Matlab;
- one of the fastest CNN implementation today [8];
- designed by Berkley vision and learning center and distributed under BSD 2 license [8, 13].

3 Segmentation algorithm

3.1 Segmentation algorithm overview

Tumor segmentation procedure based on CNN consists of preparatory stage – training of classifier on the training CT-images, and applying classifier to CT-images for segmentation. Segmentation algorithm is comprised of the following steps.

1. Pre-processing step, this step includes selection of the segmentation region using liver mask an mapping CT values from interval [-2000, 2000] to interval unsigned integer [0,255], preferable for CNN training;
2. Voxel-wise classification using pre-trained CNN, as a result of this step we obtained binary image, where 1 corresponds to tumor and 0 to healthy tissue;
3. Finally, we made smoothing and thresholding as post-processing of the obtained binary image.

3.2 CT image pre-processing on training and segmentation stages

Pre-processing of CT image applied on both training and segmentation stages. Pre-processing on training stage allows us to significantly improve segmentation quality by applying additional dividing of affected tissues. Pre-procesing stage allows us to speed up segmentation time by 6 times.

Transformation of input CT image range
DICOM data has a size of 512x512xN, where N is a number of axial slices. Each CT voxel is the 32 bit signed integer and stores a value of roentgen density on Hounsfield scale in interval from -2,000 to 2,000. Our experiments show that CNN training with this datatype is inaccurate. Histogram analysis shows that density of examined tissues

lies on interval [-60; 240]. We map this interval to [0; 255] of unsigned 8 bit integer to improve training and classification accuracy.

CT-images have different physical resolution, so called spacing. Spacing is a size of each dimension of voxel. Experiments have shown that casting to equal spacing (0.78 mm; 0.78 mm) increases quality of segmentation.

For training purposes, 2D image tiles of 64x64 pixels have been extracted from DICOM. Totally, about 4 million tiles have been extracted from the CT dataset. Image tiles of both classes were taken in ratio 1:1 in our training set to provide stable training and validation.

Preliminary dividing train data into 3 classes

Original data for training has 2-class segmentation (healthy/affected tissue). From the experiments was noticed that class of invaded tissue (tumors) has huge variance. We also noticed that tumors often consist of two kinds of tissue: dark inner tissue and peripheral lighter tissue. This fact significantly decreases segmentation quality. We manually segmented our train set into 3 classes, with dividing the class of tumor tissue into 2 subclasses. The example of such segmentation is shown in figure 1. Such modification increased average VOE in 3 times comparing on ordinal 2-class method. Detailed comparison is presented in table 1.

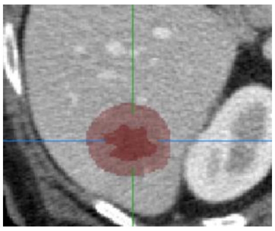

Fig. 1. Dividing tumors class on two subclasses

Performance improvement

We perform experiments on GPU NVidia Tesla K20m. These experiments have shown average execution time across the whole liver volume. It has been about 15-20 minutes. This is too long and some improvements have been necessary. In order to increase performance we have used grouping: only one voxel of the group goes to classifier while other voxels of group take their values from interpolation. Figure 2 presents quality and execution time variation with linear size of a group.

As we can see, the best quality/time ratio has been reached in groups with size of 3x3. Therewith the acceleration by 6 times has been reached in comparison with classification without grouping whereas the error (VOE) increases only by 0.6%. Finally proposed algorithms process one CT slice of typical liver in 25 seconds, for whole liver volume average processing time is 4 minutes on Nvidia Tesla K20.

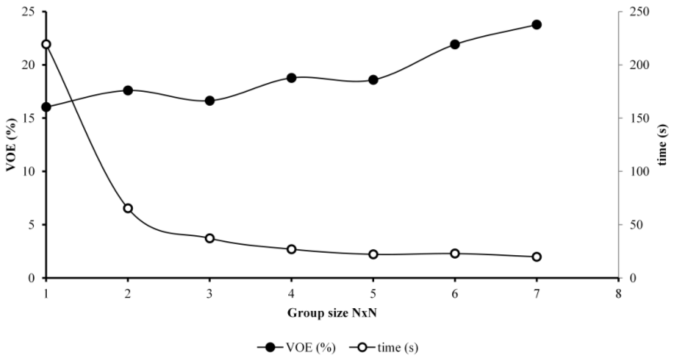

Fig. 2. Quality and execution time variation with linear size of a group

3.3 Post-processing

In this approach each pixel would be classified independently. It leads to gaps and noisiness in resulting images. An example, given in the figure 3, demonstrates that segmentation is sharp with single false voxels.

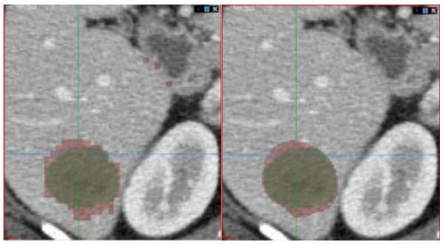

Fig. 3. Comparison of our segmentation (red), and ground true(green)
Left: before post-processing; Right: after post-processing

After classification we use 2 filters (Gaussian bluer with $\sigma = 4$ and threshold) for making our segmentation more smoothly and for removing false voxels.

4 Results

We use 5 CT images from the anonymized Samara State Medical University dataset to train and validate our classifier. Due to small size of available dataset, we used cross-validation approach to estimate generalization ability of proposed machine learning segmentation algorithm.

There are various cross-validation approaches. First is leave-one-out (LOO) – it is a quality estimation method wherein one sample could be excluded from study subset and classifier tests on it. Such procedure is performed for each sample and finally the result has been averaged [14]. The drawback of LOO is computing ineffectiveness.

In other hand there are methods based on random partitioning. It is possible to divide a set in such a way that one random subset would be excluded from training and be included in testing. This procedure can’t be repeated many times because of computing ineffectiveness. Thereby it is possible that some samples have been in test subset more often than others and estimation could become shifted.

QFold-CV – it is a compromise between LOO and random partitioning. The set has divided (by some arbitrary way) into n non-intersected blocks with equal (or nearly) length. One after another, Q random blocks become test subset while training is performed on the rest of the blocks [15].

We have used Q-Fold-CV in this work: the set has been divided into 4 blocks. The results of all CT studies after post-processing are presented in the table 2. Notice that the average VOE value in 3-class approach is 39% less compared with 2-class approach.

Table 2. The cross-validation results

CT image №	VOE for 2-class classification	VOE for 3-class classification
1	58.56%	15.17%
2	35.69%	30.56%
3	46.62%	17.04%
4	59.08%	14.46%
5	80.32%	9.42%
Average	**56.06%**	**17.33%**
STD	**14.89%**	**7.08%**

Using 3-class training allows us to reduce overfitting effect. Overfitting is one of the common problems of machine learning approaches. Overfitting (overtraining) is an undesirable fact whereby an error on training monotonically decreases while an error on test data decreases to some limit and then continues to increases [10].

On figure 1 the error on train and test subsets at two-class approach before post-processing are shown.

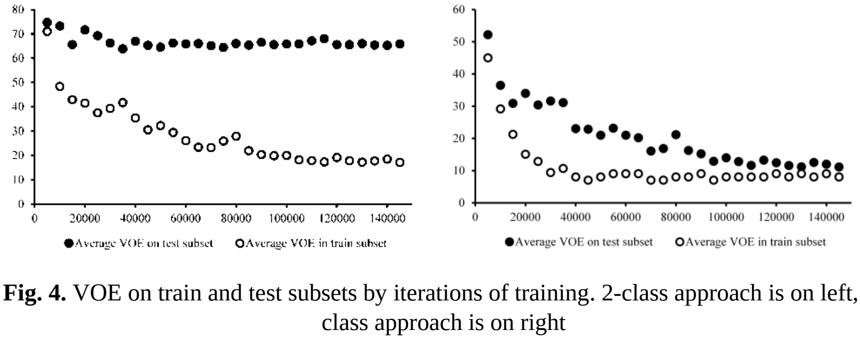

Fig. 4. VOE on train and test subsets by iterations of training. 2-class approach is on left, 3-class approach is on right

It is clear that the error on test subset reaches its minimum on 40,000 and 90,000 iteration for 2-class and 3-class approach respectively. Thus, further training leads not to the improvement of generalizing ability of the classifier but to better approximation of train data. According to [12], CNN does not susceptible to overfitting effect. In the figure 4(left part) it can be seen that in case of ordinal 2-class classification overfitting is observed partially: while the error on train subset decreases, the error on test subset is fixed and doesn't increase any longer. Proposed 3-class approach allows to exclude overfitting effect (fig. 4, right part).

5 Conclusion

CNN shows the ability to detect liver tumors on an equal basis with other methods. In the present work we shown that AlexNet-like architecture is applicable for solving liver tumors segmentation problem [11]. We have analyzed method's quality in pre- and post-processing and have stated the fact of overfitting (partly). We have fixed the number of iteration for optimal quality on test data. We've also made performance improvements that boost execution by 6 times with negligible decrease of quality.
The main contribution is additional dividing into subclasses, what lets significantly (by 39% VOE) increase segmentation quality. Thus, we have the quality that is comparable with analogous works [6]. The objective of future researches is to create and develop methods for automatic dividing classes into subclasses, for instance, clustering.

Acknowledgements

The work was performed as part of the project "Autoplan" (automatic system of planning and monitoring of the operation) under the state contract to perform research and development work "Development of technology and organization of production of the automated systems of planning, management and monitoring of results of surgical treatment" ("4.3-Autoplan-2014").

References

1. Pham M, Susomboon R, Disney T, Raicu D, Furst J. A comparison of texture models for automatic liver segmentation. Proc. SPIE, 2007; 6512.
2. Toro J, Oscar A, Foncubierta-Rodríguez A, Depeursinge A, Müller H. Texture classification of anatomical structures using a context–free machine learning approach. Proc. SPIE, 2015; 9114.
3. Depeursinge A, Foncubierta-Rodriguez A, Ville D, Müller H. Three-dimensional solid texture analysis in biomedical imaging: Review and opportunities Medical Image Analysis, 2014; 18: 176-196.
4. Gilmanov T. A comparison of segmentation methods for road signs detection problem. Science Time, 2015; 1(13): 86-91.

5. Roth H, Farag A, Lu L, Turkbey B, Summers R. Deep convolutional networks for pancreas segmentation in CT imaging. CoRR, 2015; 1504; 03967.
6. Vivanti R. Automatic liver tumor segmentation in follow-up CT studies using Convolutional Neural Networks Proc. Patch-Based Methods in Medical Image Processing Workshop, MICCAI, 2015.
7. Qi Y, Xiong Leow W, Tian Q, Zhou J, Han T, Venkatesh S, Wang S. Semi-automatic Segmentation of Liver Tumors from CT SCans Using Bayesian Rule-based 3D.The MIDAS Journal, 2008.
8. Framework Caffe. URL: http://caffe.berkeleyvision.org (accessed 1 Feb 2016).
9. LeCun Y, Bengio Y. Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 1995; 3361; 10.
10. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning, 2nd edition. Springer, 2009; 533 p.
11. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012: 1097-1105.
12. 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, 2014; 15(1): 1929-1958.
13. Caffe ModelZoo. URL: http://caffe.berkeleyvision.org/model_zoo.html (accessed 1 Feb 2016).
14. Gelfand AE. Model determination using sampling-based methods. Markov chain Monte Carlo in practice, 1996: 145-161.
15. Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection Ijcai, 1995; 14(2): 1137-1145.