Usage of fully convolutional neural network for automation of extracting the left ventricle contour on the ultrasonic data images

Andrey A. Mukhtarov[1], Vasiliy V. Zyuzin[1] and Anastasia O. Bobkova[1]

Ural Federal University, Yekaterinburg, Russia
andrew443209993@yandex.ru,iconismo@gmail.com,zvvzuzin@gmail.com

Abstract. The article discusses experience of application of fully convolutional neural networks for automation of left ventricle contouring. Results of the quality analysis of contouring show that this approach can be used to automate the work of cardiologists with the echographic data.

Keywords: Contouring, left ventricle, neural networks, ultrasonic images, image segmentation

1 Introduction

Cardiologists use the echographic data of patients to determine the left ventricle (LV) area of the heart in order to study the contractility of the left ventricle walls, restore the LV volume, and calculate various indicators. As a rule, the contour is selected subjectively, and it depends on qualification of the physician performing the procession of medical images. Such diagnostics takes a long time and is not always accurate.

At the moment, there are no automated software tools that allow one to fully automate the LV contouring on the heart ultrasonic data. Thus, the problem of increasing the speed and quality of diagnostics by automating the LV contouring is actual.

2 Choice of neural network model

To solve the problem, it was decided to use some machine learning method i.e., the neural networks. The results of literature research on the similar subjects show that a fully convolutional neural network (FCN) gives the best results for the problem of image segmentation. This network is similar to convolutional neural network (CNN) where the last fully connected layer is replaced by another convolution layer with a large susceptible field. The idea is to capture the global scene context, which gives information about objects on the image including their localization.

Analysis of existing studies of similar problems showed that the neural network AlexNet is the most popular implementation of the CNN for the general classification of objects. The AlexNet model outperforms competing approaches based on traditional functions in solving a number of computer vision problems.

Nox existing approach is based on the recent successful results of using the deep networks for the image classification [1–3]. Chen Liang-Chieh, Evan Shelhamer, and Phi Vu Tran [4–6] presented the experience of using fully convolutional neural networks. Moreover, authors of article [5] described in details the solution of the multiclass semantic segmentation problem. Our paper describes how the CCN model of the AlexNet is converted to FCN-32 in order to be able to train the network using images of arbitrary size. Furthermore, the authors show how to tune maximally fine the neural network for the best classification by converting the network to FCN-8. The experiments were carried out on Pascal VOC data, which include color images of various types. By their research, it was shown that such application of the fully convolutional neural networks gives the best result.

For the presented problem, it was decided to use the pre-trained model FCN-8-AlexNet-pascal. Since in our case just only one object is required to be identified, there should be only two classes at the network outlet: the background and the LV region. Therefore, an additional convolutional layer with two outputs was added to the source network. In order to determine that the network has started to relearn at an early stage, another set of images is used in training whith volume about 10% of the training set. The network is not training on this test set. Network predicts the result on the test set, and a test error is determined. If an error on the training set is usually reduced, then the test error can increase in this case. This means that the network has become more receptive to its training set and the rest part of the images will not be recognized correctly. So, it is necessary to change the training parameters, network structure, or training set.

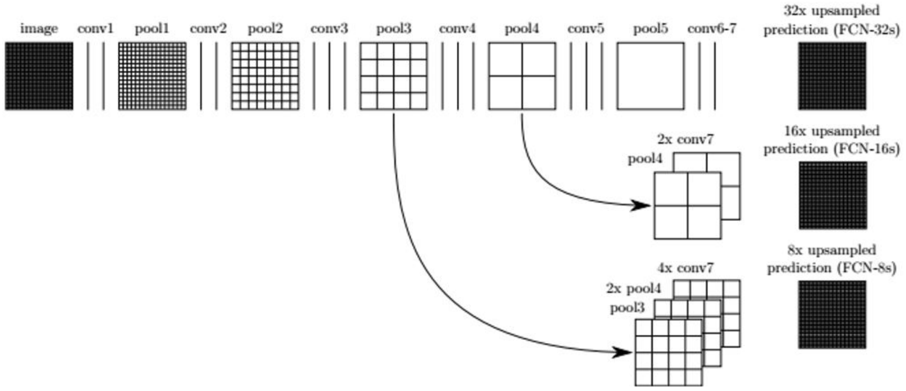
image
conv1
pool1
conv2
pool2
conv3
pool3
conv4
pool4
conv5
pool5
conv6-7
32x upsampled prediction (FCN-32s)
2x conv7
pool4
16x upsampled prediction (FCN-16s)
4x conv7
2x pool4
pool3
8x upsampled prediction (FCN-8s)

Fig. 1. FCN-32 network transformation to FCN-8

Figure 1 shows several intermediate consecutive convolutional layers (vertical lines) for highlighting more complex maps of image feature. The pool layers are indicated by a grid that shows the relative spatial dimension. The first line (FCN-32) is a single-stream network that allows one to predict an area of 32 pixels in one step. The second line (FCN-16) allows predicting the network more subtle details while retaining the high-level semantic information by combining the forecasts of the last layer and the pool4 layer (areas of 16 pixels). The third line (FCN-8) provides additional classification accuracy from the pool3 projections.

A convolutional layer has a set of matrix filters that are applied to images and determine the feature. A combination of such several layers will build new features according to the previous features of a lower order. In practice, this means that the network is trained to see complex features, which are a composition of simpler ones.

The pooling layer represents a layer without training. Here, the images are filtered highlighting the largest value of the pixel in the area and ignoring the others. Thus, the image decreases in size and the most significant features are left regardless of their location.

The next three layers are a fully connected network. Here, each neuron takes in the input all the outputs of the previous layer neurons. Then, the upsample layer performs the image increase.

Thus, it was decided to use the original FCN-8 model with some modifications. To avoid overfitting of the network, it was decided to add layers of normalization and dropout. Dropout randomly disconnects some neurons from the fully connected layer during training.

3 Building a model

The training was conducted on ultrasound images of patients, the total number included 1895 images. From them 90% of the frames were used as a training sample, and the remaining 10% as the test set. The study was conducted on a GPU (graphics processing unit) using the caffe framework and the nVidia GTX 1070 graphics card. The training lasted for 12 hours and was fulfilled by the backpropagation method. Classification is based on blocks of pixels, i.e., the central pixel and 8 nearest to it.

The following steps are necessary to train the neural network. The training takes place through several iterations. Their number is set during the training of the network. The network passes through all input data at each iteration. The following steps are performed at each iteration:

1. upload the data and initialize the weights in random order;
2. perform the direct propagation;
3. calculate losses;
4. perform the reverse propagation;
5. update the weights using gradient descent;
6. repeat from step 2 until all the iterations run out.

The loss function is a mathematical function (with the current set of parameters) that shows the quality of classification. The selected pretrained model was used to select simultaneously several objects in the images. In order to classify each pixel of the image, a map of all detected objects in the image was used. Each pixel has an assosiated label. In order to differ the objects, the colors were indexed, so, each object had its own color. Therefore, the training of the network required preliminary processing of the input data. For this purpose, the images with expert contours were converted to RGB with a mask for color indexing.

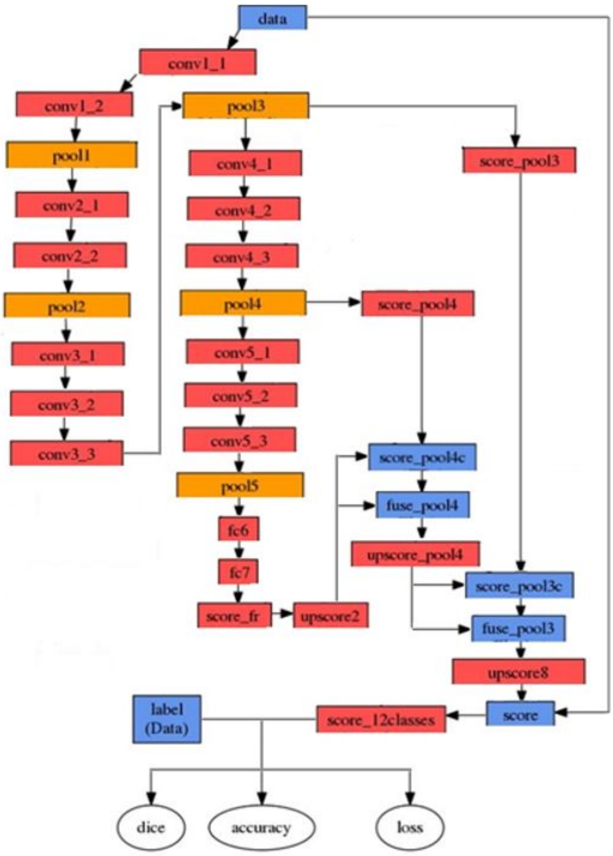

Fig. 2. Architecture of the neural network

The network architecture is presented in Table 1 and in Fig. 2; examples of the input data are shown in Fig. 3.

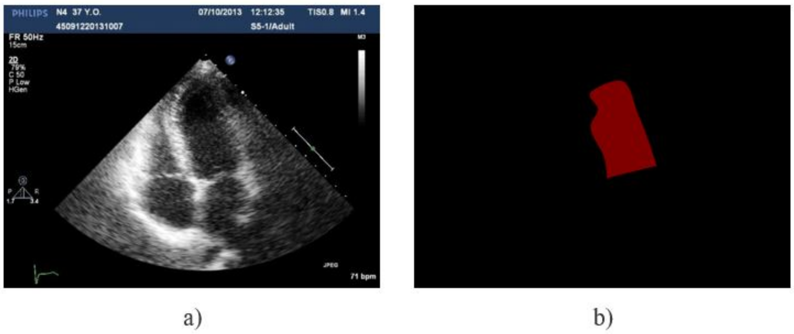
PHILIPS N4 37 Y.O. 07/10/2013 12:12:35 TIS0.8 MI 1.4
45091220131007 S5-1/Adult
FR 50Hz
15cm
71 bpm
a)
b)

Fig. 3. Input data; a) ultrasound image; b) expert contour

Table 1. Neural network architecture

Layer	Name	Number and size of cards of features	Core size
0	Data	1 x 600 x 800	
1	Conv1_1	64 798 x 998	3x3
2	Conv1_2	64 798 x 998	3x3
3	Pool1(MAX Pooling)	64 399 x 499	2x2
4	Conv2_1	128 399 x 499	3x3
5	Conv2_2	128 399 x 499	3x3
6	Pool2(MAX Pooling)	128 200 x 250	2x2
7	Conv3_1	256 200 x 250	3x3
8	Conv3_2	256 200 x 250	3x3
9	Conv3_3	256 200 x 250	3x3
10	Pool3(MAX Pooling)	256 100 x 125	2x2
11	Conv4_1	512 100 x 125	3x3
12	Conv4_2	512 100 x 125	3x3
13	Conv4_3	512 100 x 125	3x3
14	Pool4(MAX Pooling)	512 50 x 63	2x2
15	Conv5_1	512 50 x 63	3x3
16	Conv5_2	512 50 x 63	3x3
17	Conv5_3	512 50 x 63	3x3
18	Pool5(MAX Pooling)	512 25 x 32	2x2
19	Fc6	4096 19 x 26	7x7
20	Fc7	4096 19 x 26	1x1
21	Score_fr	21 19 x 26	1x1
22	Upscore2	21 40 x 54	4x4
23	Score_pool4	21 50 x 63	1x1
24	Score_pool4c	21 40 x 54	
25	Fuse_pool4	21 40 x 54	
26	Upscore_pool4	21 82 x 110	4x4
27	Score_pool3	21 100 x 125	1x1
28	Score_pool3c	21 82 x 110	
29	Fuse_pool3	21 82 x 110	
30	Upscore8	21 664 x 888	16x16
31	Score	21 600 x 800	
32	Score_12classes	2 600 x 800	1x1

4 Evaluation results

To quantify the quality of contouring, it was decided to use the following criteria:

- precision

$$Precision = \frac{S_{\cap}}{S_{cont}}, \tag{1}$$

  where $S_{\cap}$ is the intersection of the square area limited by expert contour and area formed from classified pixels, S_{cont} is the square area formed from the classified pixels;
- recall

$$Recall = \frac{S_{\cap}}{S_{exp}}, \tag{2}$$

  where S_{exp} is the area of the region bounded by the expert contour;
- F-measure

$$F = \frac{2 * Precision * Recall}{Precision + Recall}; \tag{3}$$

- proportion of erroneously classified pixels;
- proportion of correctly classified pixels;
- area under receiver operating characteristic curve (AUC).

The results of network training are shown in Fig. 4.

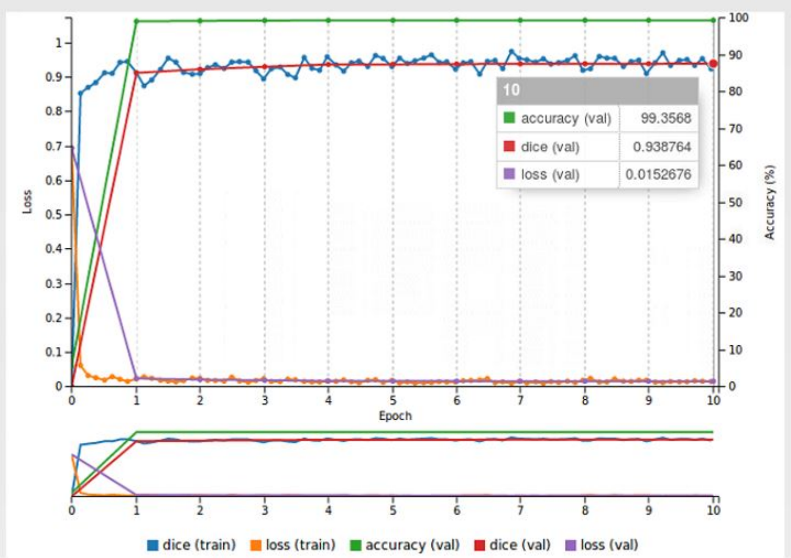

Fig. 4. The result of learning the network FCN-8; dependence of losses and classification accuracy on the number of iterations

Figure 4 shows that the model has been trained quite well. The losses of the training and test samples are close to zero, while the dice (the Sorensen coefficient) on the test sample reaches the value of 94%.

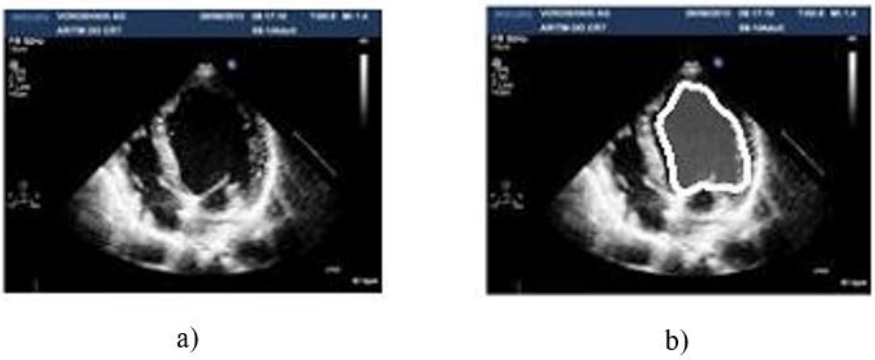

Fig. 5. Results of contouring; a) ultrasound image; b) extracted contour of LV

Figure 5 shows an example of contour determination using the trained network.

Table 2 compares the results of the method of neural networks with the results of contouring by the decision tree method and the ensemble of trees [7, 8].

Table 2. Comparison of automatic contouring methods

Criterion	Recall	Precision	F	Overall Accuracy, %	Overal Error, %	AUC
Neural networks	0.97±0.05	0.91±0.06	0.94	99.27	0.73	0.99
Decision tree	0.77±0.01	0.92±0.02	0.84	94.6	5.4	0.96
Ensembles of trees	0.78±0.01	0.97±0.02	0.86	98.4	1.6	0.99

Table 2 shows that neural networks give the best result of the quality of LV contour determination on ultrasound images.

5 Conclusion

Results of the research show that the method of fully convolutional neural networks can be used to distinguish the LV heart contour on ultrasound data. This method gives the best results in comparison with other researched methods. To increase the accuracy of contouring, the increase in the train sample is required, and the use of other neural networks models is also possible.

References

1. Krizhevsky, A., Sutskever, I., Hinton, G. E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems. 1097–1105 (2012)
2. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. (2014)
3. Szegedy, C. et al.: Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition, 1–9 (2015)
4. Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. International conference on learning representations. (2015)
5. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. Computer vision and pattern recognition. 39, 640–651 (2017)
6. Tran, P. V.: A fully convolutional neural network for cardiac segmentation in Short-Axis MRI. Computer vision and pattern recognition. (2016)
7. Zyuzin, V. V., Bobkova, A. O., Porshnev, S. V., Mukhtarov, A. A., Bobkov, V. V.: The application of decision trees algorithm for selecting the area of the left ventricle on echocardiographic images. First International Workshop on Pattern Recognition, 10011, 100110I-1 – 100110I-7 (2016)
8. Porshnev, S. V., Mukhtarov, A. A., Bobkova, A. O., Zyuzin, V. V., Bobkov, V. V.: The study of applicability of the decision tree method for contouring of the left ventricle area in echographic video data. CEUR Workshop Proceedings, 1710, 248–258 (2016)