

Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images

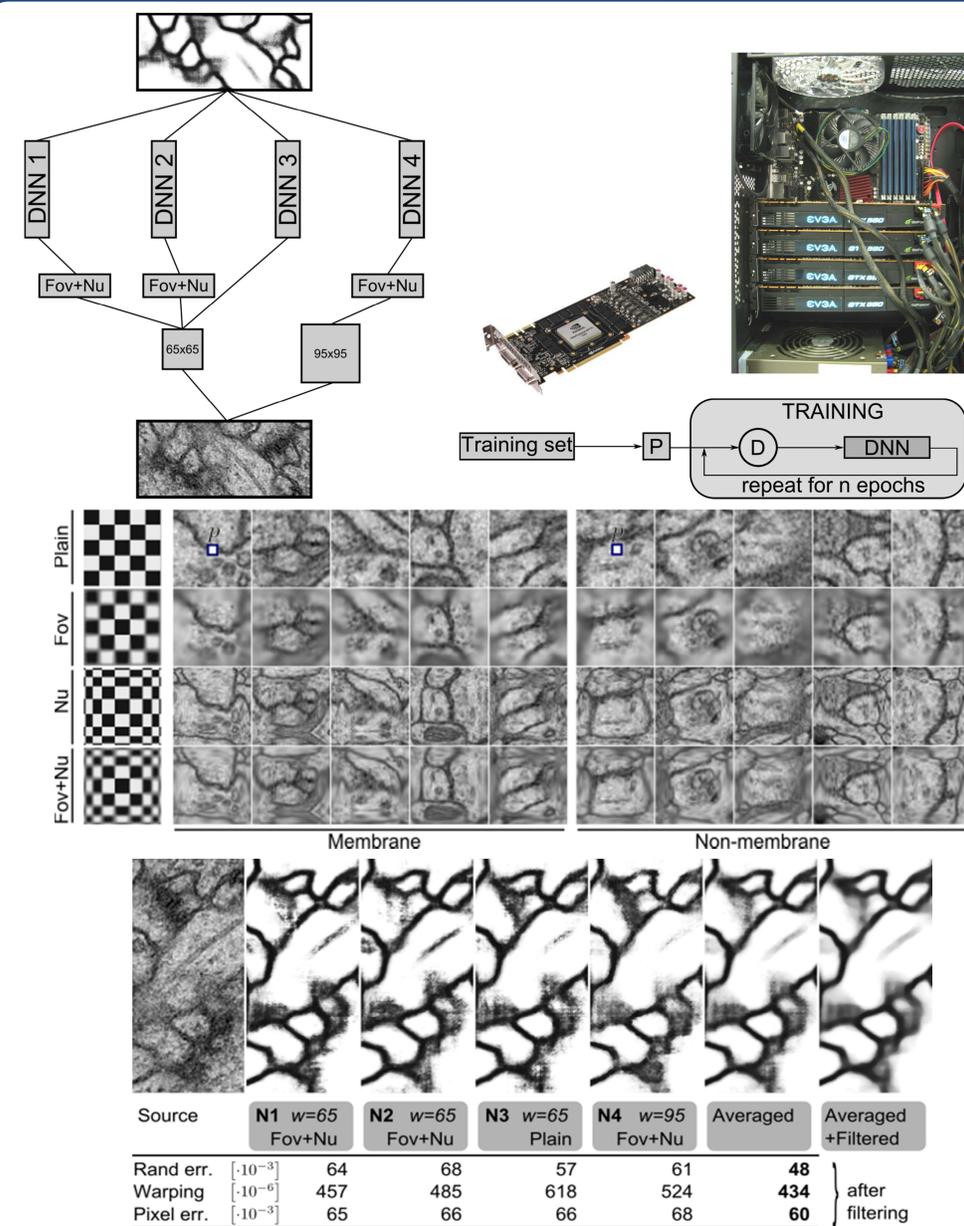
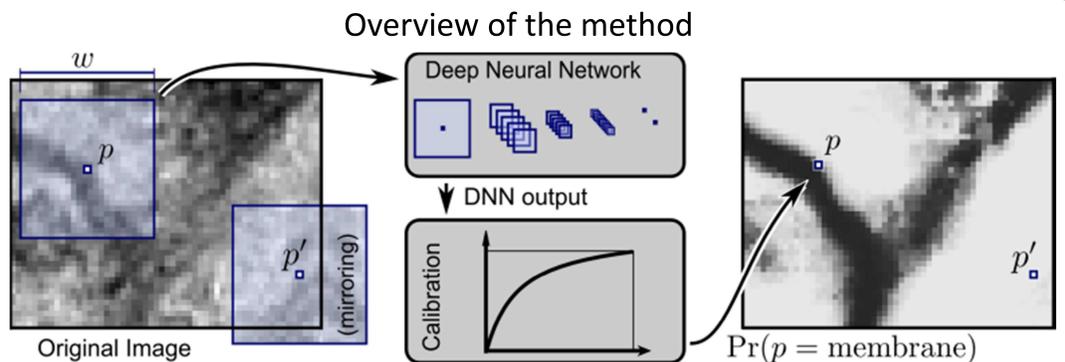
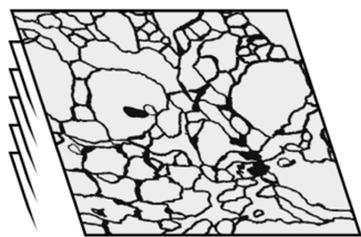
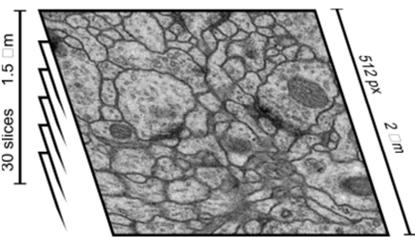
Dan C. Cireşan, Alessandro Giusti, Luca M. Gambardella, Jürgen Schmidhuber

IDSIA, USI and SUPSI, Manno-Lugano, Switzerland

{dan, alessandro, luca, juergen}@idsia.ch

We describe the method that won the ISBI 2012 Electron Microscopy (EM) Segmentation Challenge. The automatic segmentation of neuronal structures in stacks of EM images is necessary to efficiently map 3D brain structure and connectivity. To segment biological neuron membranes, we use a special type of deep artificial neural network as a pixel classifier. The label of each pixel (membrane or nonmembrane) is predicted from raw pixel values in a square window centered on it. The input layer maps each window pixel to a neuron. It is followed by a succession of convolutional and max-pooling layers which preserve 2D information and extract features with increasing levels of abstraction. The output layer produces a calibrated probability for each class. The classifier is trained by plain gradient descent on a 512x512x30 stack with known ground truth, and tested on a stack of the same size (ground truth unknown to the authors).

Training data: 30 labeled 512x512 slices
Testing data: 30 unlabeled 512x512 slices



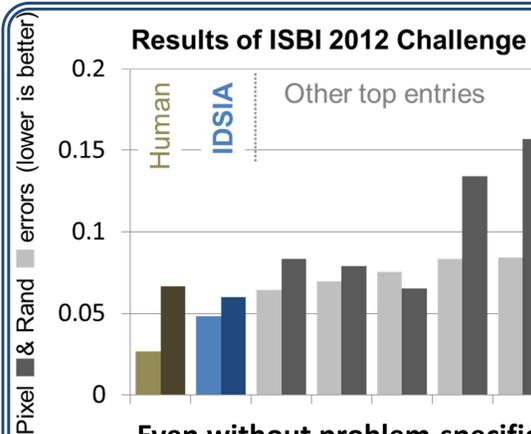
- 4 x GTX 580 1.5GB RAM, more than 6 TFLOPS (maximum theoretical speed)
- 50x speed-up compared with a single threaded CPU version of the DNN program (one day on GPU instead of two months on CPU)
- Training time for one epoch varies from approximately 170 minutes for DNN1-DNN3 ($w = 65$) to 340 minutes for DNN4 ($w = 95$). All nets are trained for 30 epochs, which leads to a total training time of several days
- Testing the trained networks on new images is relatively fast: classifying the 8 million pixels comprising the whole testing stack takes 10 to 30 minutes on four GPUs

Deep Neural Network Architecture (DNN4)

Layer	Type	Maps and neurons	Kernel size
0	input	1 map of 95x95 neurons	
1	convolutional	48 maps of 92x92 neurons	4x4
2	max pooling	48 maps of 46x46 neurons	2x2
3	convolutional	48 maps of 42x42 neurons	5x5
4	max pooling	48 maps of 21x21 neurons	2x2
5	convolutional	48 maps of 18x18 neurons	4x4
6	max pooling	48 maps of 9x9 neurons	2x2
7	convolutional	48 maps of 6x6 neurons	4x4
8	max pooling	48 maps of 3x3 neurons	2x2
9	fully connected	200 neurons	1x1
10	fully connected	2 neurons	1x1

- 10 layers (deep) of non-linear neurons, with tens of maps per layer (large), loosely inspired from biology
- Introduced by Fukushima (1980) and refined by LeCun et al. (1998), Riesenhuber et al. (1999), Simard et al. (2003), Behnke (2003), Cireşan et al. (2011)
- Fully supervised, with randomly initialized filters, trained minimizing the misclassification error
- Hard/impossible to train with stochastic gradient descent two decades ago
- Neither supervision nor carefully prewired synapses are needed
- Architecture:
 - convolutional layer (extracts features), max-pooling layer (selects features)
 - one output neuron per class, normalized with soft-max activation function

Results of ISBI 2012 Challenge



Group	Rand error [$\cdot 10^{-3}$]	Warping error [$\cdot 10^{-6}$]	Pixel error [$\cdot 10^{-3}$]
Second Human Observer	27	344	67
Simple Thresholding	445	15522	222
Our approach	48	434	60
Laptev et al. [24] (1)	65	556	83
Laptev et al. [24] (2)	70	525	79
Sumbul et al.	76	646	65
Liu et al. [26] (1)	84	1602	134
Kaynig et al. [23]	84	1124	157
Liu et al. [26] (2)	89	1134	78
Kamentsky et al. [20]	90	1512	100
Burget et al. [8]	139	2641	102
Tan et al. [36]	153	685	88
Bas et al. [4]	162	1613	109
Iftikhar et al. [18]	230	16156	150

Conclusions

- A general technique for bio-medical image segmentation
- Won three other competitions:
 - ICPR 2012 Mitosis Detection in Breast Cancer Histological Images
 - ICDAR 2011 Offline Chinese Character Recognition
 - IJCNN 2011 The German Traffic Sign Recognition Benchmark
- Future work
 - Applying the method to other bio-medical problems
 - Preliminary tests with dynamic programming show three orders of magnitude speed-up when applying the nets

Acknowledgment

This work was partially supported by Swiss National Foundation grant 200020-140399/1.

More info

<http://www.idsia.ch/~ciresan>



Even without problem-specific postprocessing, our approach outperforms competing techniques by a large margin in all three considered metrics, i.e. rand error, warping error and pixel error. For pixel error, our approach is the only one outperforming a second human observer.