Proceedings of the 35th Chinese Control Conference
July 27-29, 2016, Chengdu, China

Brain MRI Segmentation with Patch-based CNN Approach

Zhipeng CUI, Jie YANG, Yu QIAO*
Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University
The Key Laboratory of Ministry of Education for System Control and Information Processing, China

Abstract:
Brain Magnetic Resonance Image (MRI) plays a non-substitutive role in clinical diagnosis. The symptom of many diseases corresponds to the structural variants of brain. Automatic structure segmentation in brain MRI is of great importance in modern medical research. Some methods were developed for automatic segmenting of brain MRI but failed to achieve desired accuracy. In this paper, we proposed a new patch-based approach for automatic segmentation of brain MRI using convolutional neural network (CNN). Each brain MRI acquired from a small portion of public dataset is firstly divided into patches. All of these patches are then used for training CNN, which is used for automatic segmentation of brain MRI. Experimental results showed that our approach achieved better segmentation accuracy compared with other deep learning methods.

Key Words: CNN, Deep Learning, Brain MRI Segmentation, Patch-based

1 Introduction

The segmentation of brain MRI has been a hot area of computer vision for several years. Segmentation serves as an important step for quantitative analysis in brain MRIs and for the research of brain disorders. Indeed, structural variation in the brain may cause some brain disorders. Quantification of structural variation by measuring volumes of region of interest, can be used to evaluate severity of some disease or evolution in brain [1]. Only when the labeling is processed on MRI, these measurements can be performed. Segmentation of MRI plays an increasingly important role in medical image processing and analysis as digital medical image developing. Thousands of segmentation methods were developed, which are mainly edge-based and contour-based [2]. However, with these methods, it is a very hard task for segmenting complex structure of medical image with high accuracy rate [3].

Deep learning is a type of machine learning approaches, which arise from artificial neural network. David Rumelhart, Geoffrey Hinton and other individuals applied backpropagation algorithm to artificial neural network, which started machine learning based on statistic model [4]. Artificial neural network was limited to complex structure and heavy training time. Neural networks reappeared as deep learning which could learn feature hierarchy with the development of hardware in 2006. The purpose of deep learning is to learn multiple levels of representation and find interesting structure in data [5]. Modern deep learning methods can represent functions of increasing complexity as the layer is added. The way of processing data in deep learning is similar to human brain. Deep learning has made great progress in these years. It has been applied to object recognition tasks in ImageNet and feature learning from unlabeled data [6]. Multiple levels of representation and underlying distribution of the data can be automatically learned with deep learning [7].

Convolutional Neural Networks (CNNs) are a type of fully trainable models with multi-layer [8]. CNNs are biologically-inspired variants of MLPs derived from Hubel-Wiesel model, which are successful in visual processing algorithms. The CNNs are a variety of deep learning methods, which can learn a deep feature hierarchy from images [9]. CNN has advantages on processing images whose training data is not limited to 1D. Training data of CNN can be 1D acoustic data, 2D image data or 3D video data. The hidden layers of a CNN consist of *convolutional layers* and *pooling layers* [10]. *Feature maps* of the layer represent the number of features extracted from preceding layer. *Filters* in the layer are used to process feature maps passed from former layer. Filters are identical to the number of feature maps.

There are quite a few limitations in brain MRI segmentation. Segmenting brain MRI with traditional methods is time-consuming and requires prior medical knowledge. In addition, training data is another major concern in brain MRI. It is usually hard to collect brain MRI. To overcome the difficulties in brain MRI segmentation, we implemented a patch-based CNN with time cost sufficiently low. The proposed CNN outperforms other CNNs with different structures and ANNs in segmentation accuracy. Patch-based CNN has well solved an insufficient amount of training data.

2 Method

This paper used the data from a public dataset, which can be download at CANDI neuroimaging access point to conduct the experiment. Jean A. Frazier , et. al. manually segmented MRIs in this dataset [11]. It comprises 103 MRIs from four diagnostic groups: Bipolar Disorder with Psychosis, Bipolar Disorder without Psychosis, Schizophrenic Spectrum and healthy control [12]. The subjects are from the 6 to 17age group, including children and adolescents, female and male. All of the images were recruited at the McLean Hospital Brain Imaging Center on a 1.5-Tesla magnetic resonance scanner (General Electric Signa Scanner) [13].

2.1 Data Preprocessing

In this paper, we extracted a few small sets of MRIs from the dataset randomly, each set consists of 4 to 5 MRIs. We divided MRIs whose size is 256×256 to 32×32 and 13×13 patches according to the label on each pixel. Quiet a few patches extracted from brain MRI are useless due to imaging modality. About 25000 of 65536 patches are left as our training set. Each pixel is marked by the label of central pixel in each 32×32 or 13×13 patch. Training set contains about 100000 patches used to train networks.

* Corresponding author: Yu QIAO, qiaoyu@sjtu.edu.cn

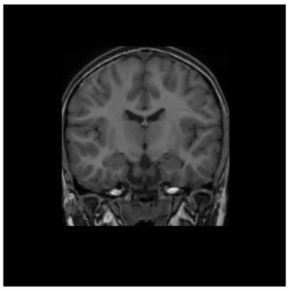


Fig. 1: Raw brain MRI, which was acquired at the McLean Hospital Brain Imaging Center on a 1.5 Tesla General Electric Signal Scanner.

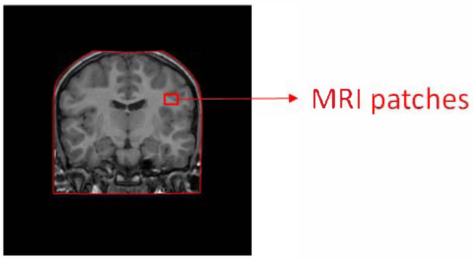


Fig. 2: Segmentation of brain MRI into patches, stride is 1 pixel

2.2 CNN architecture

Convolutional layer

In a convolutional layer, feature maps in previous layer are convolved learnable kernels [14]. The convolutional layer convolves the feature maps, sums the responses, adds a bias term, which can obtain feature in each region of images. Images will reduce in dimensionality after convolving. Then the activation function act on the values acquired from kernels. Output maps are combined with multiple convolved feature map. The output in convolutional layer of each kernel is denoted as:

$$y_j = b_j + \sum_i k_{ij} * x_i \tag{1}$$

In formula (1), k_{ij} denotes the filter for the jth feature map, and x_i denotes input, b_j denotes bias of filter.

Pooling layer

A pooling layer produces down-sampled feature maps of previous layer. The number of feature maps will remain unchanged in pooling layer. In this paper, we used $maxpooling$ operation in pooling layers , which takes the maximum value in $n \times n$ non-overlapping region [15]. Maxpooling operation is described as:

$$y_j = \beta_j maxpooling(x_j) + b_j \tag{2}$$

In formula (2), β_j denotes coefficient of pooling.

ReLU Activation function

Activation function is usually added after convolutional layer to speed up training proceed. Activation won't change size of feature maps but will change values of input feature maps. In this paper, we used $ReLU$ function as our activation function, which is denoted as:

$$\sigma(x) = max(0, x) \tag{3}$$

$ReLU$ function has its unique advantages over other activation functions. In 2001, Dayan, P and Abbott simulate common neural activation function motivated by biological data [16]. This activation function can be approximated by $ReLU$ function. $ReLU$ function is one-sided and there won't exist antisymmetry or a sign symmetry [16]. It allows a network to obtain sparse representation in neural networks and can dramatically shorten training time.

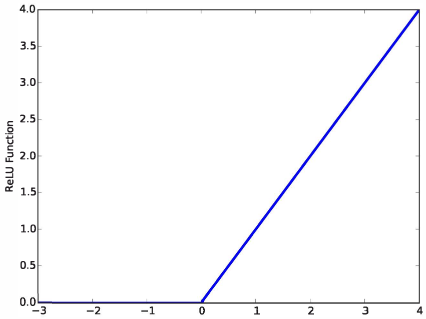


Fig. 3: Rectified linear unit function in CNN

Softmax Regression

Softmax regression [19] is chosen as classifiers in our classification tasks after fully-connected layers. Softmax Regression is denoted as follow equation:

$$h_\theta(x^i) = [p(y^i = 1 \mid x^i; \theta), p(y^i = 2 \mid x^i; \theta)$$

$$...p(y^i = k \mid x^i; \theta)]^T =$$

$$\frac{1}{\sum_{j=1}^{k} e^{\theta_j^T x^i}} [e^{\theta_1^T x^i}, e^{\theta_2^T x^i} ... e^{\theta_k^T x^i}]^T \tag{4}$$

Segmentation of Brain MRI

In this paper, we will apply CNN for pixel-based automatic segmentation of brain MRI [15]. The same preprocess will be applied to the test brain MRI. Useful patches will be left to pass the CNNs. Each patch will get a label through CNNs. Patches and their corresponding labels will make up a new brain MRI with labels. Labels will be used to segment corresponding regions [17].

2.3 Patch-based CNN

An overview of CNN structure is shown in Fig. 5. The CNN is trained with brain MRI patches as input. Each label of the patch is determined by the center pixel. The stride in dividing patches is set as one pixel. There will be partially overlapping in adjacent patches as illustrated in Fig. 4. We implemented our CNN with MatConvNet [20] on MATLAB.

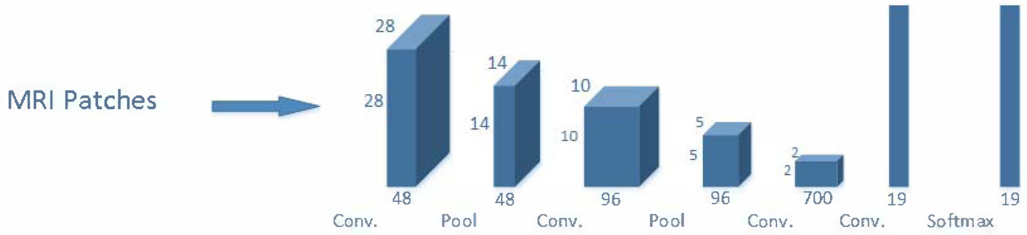


Fig. 5: Proposed 7-layer CNN. The network consists of four convolutional layers, 2 max-pooling layers and a fully-connected layer.

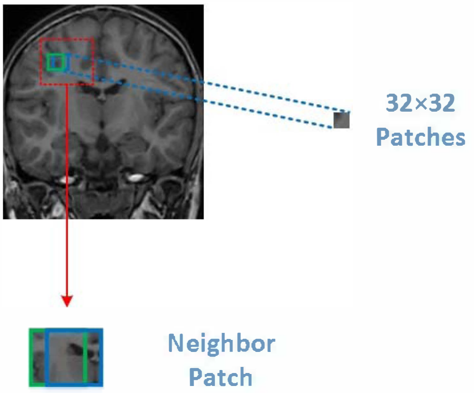


Fig. 4: A demonstration of segmentation of origin brain MRI. 32×32 patches are the input of CNN.

Table 1: Parameters of layers in CNN. Conv. denotes convolutional layer.

Proposed CNN	Layer type	No. of feature maps	Input size
layer1	conv.	48	32×32
layer2	pool	48	28×28
layer3	conv.	96	14×14
layer4	pool	96	10×10
layer5	conv.	700	5×5
layer6	conv.	19	2×2
layer7	softmax	19	1×1

Table 2: Parameters of layers in CNN1 and CNN2.

CNN1&2	Layer type	No. of feature maps	Input size
layer1	conv.	20	32×32
layer2	pool	20	28×28
layer3	conv.	50	14×14
layer4	pool	50	10×10
layer5	conv.	500	5×5
layer6	conv.	19	2×2
layer7	softmax	19	1×1

Our CNN contains 7 main layers. First layer of CNN is a convolutional layer with a kernel of size 5×5 and generates 48 feature maps. Following the first convolutional layer is a max-pooling layer whose scale is 2 to down-sampling. These two layers are used to detected low level features in patches. Next 2 layers are followed by a convolutional layer and a max-pooling layer with same kernels and generate 96 feature maps which are used to detect higher level image structures. Then another convolutional layer with 700 feature maps is applied, whose kernel size is set to 4×4. The rest of the CNN consist of a convolutional layer and a fully-connected layer. We achieve classification of image patches with $Softmaxregression$ in fully-connected layer. $ReLU$ function is used as the activation function in each convolutional layer. Size of input patches is 32×32. Dimensional variation of patches and other details are showed in Table 1. In our experiment, connecting weights are initialized from $N(0, 0.01^2)$ normal distribution randomly. Bias of each term begins at 0. Learning rate is set as 0.001. Momentum parameter of 0.9 is used and weight decay is 0.005. Dropout probability is set to 0.5 which is used to prevent over-fitting. This 7-layer CNN is set to this structure due to the control of computational cost in patch-wise segmentation framework.

Brain MRI patches are selected as input such that we can achieve sufficient classification performance and patches provide sufficient classification performance but reduce computational cost dramatically.

3 Experiment Results and Discussion

3.1 Comparison

The proposed CNN is compared against five other deep learning approaches, including three different CNN and two Artificial Neural Networks (ANNs). Three CNNs with different structures are trained to verify the influence of input's size and structure variation on segmentation accuracy. These CNNs are denoted as CNN1, CNN2 and CNN3. Input patch of CNN1, CNN2 is 32×32, and 13×13 for CNN3. Convolutional and pooling layers are in alternating order of first 3 CNNs. There is no pooling layer in CNN3 due to the size of input patch.

Layer structure and input size of CNN1 and CNN2 are identical to proposed CNN. CNN1 and CNN2 have fewer feature maps. The number of feature maps are equal in these two CNNs. $ReLU$ function is replaced by $sigmoid$ function in each convolutional layer. Details of CNN1 and CNN2 are listed in Table 2. In CNN3, we trained the network with the input size of 13×13. Max-pooling layer is not applied to CNN3. CNN3 consists of 4 convolutional layers and a fully-connected layer. The size of kernels in convolutional layers

Table 3: Parameters of layers in Proposed CNN.

CNN3	Layer type	No. of feature maps	Input size
layer1	conv.	40	13×13
layer2	conv.	160	9×9
layer3	conv.	500	5×5
layer4	conv.	19	2×2
layer5	softmax	19	1×1

of CNN3 is as follow: 5×5, 5×5, 4×4, 2×2. $ReLU$ function is applied in convolutional layers. Tabel 3 is detail of CNN3. Structures of two ANNs are illustrated in Table 4. 200000 patches from 10 MRIs are used for training set in our experiments. CNNs are trained with 80 epoches and ANNs are trained 100 epoches. Patch-based input occupy less memory and require far less training time on the same GPU. It costs about 12 hours to train the proposed CNN on Nvidia Quadro K1000M. Training error curve of proposed CNN and CNN3 are showed in following figures.

Table 4: Parameters in ANNs

	Structure	Size of patches
Neural Network 1	3(1024-150-10)	32×32
Neural Network 2	5(1024-800-400-150-10)	32×32

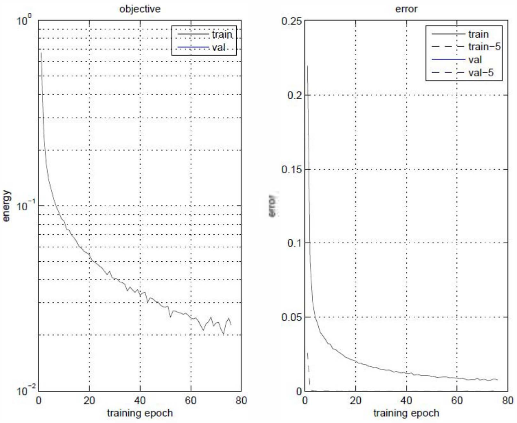


Fig. 6: Training error curve of proposed CNN

It is observed that training error decrease gradually as the epoches increase. Training error is approaching 0, which means that parameters were set correctly. CNN1 and CNN2 are different from the proposed CNN in the number of feature maps. Their training error curves are remarkably similar. Training error is trending to zero after 80 epoches. So these three CNNs converges under the training parameters. On the contrary, there exists obvious fluctuations in CNN3's training error curve. It suggests that features extracted from smaller input patch carry less information and it's harder for label learning. A CNN's learning capacity decreases due to the absence of down-sampling layer.

3.2 Visualization of CNN features

Partial weights in the first convolved layer of proposed CNN are visualized to observe properties of CNNs, which are low level features of brain MRI.

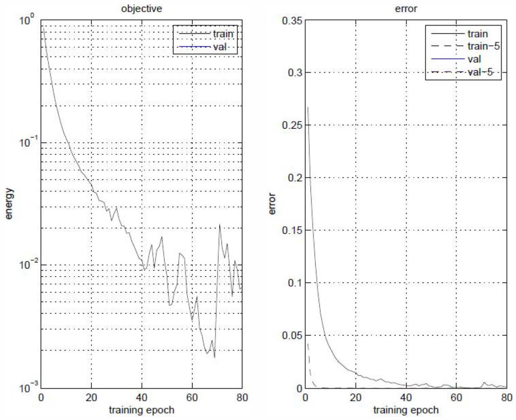


Fig. 7: Training error curve of CNN3

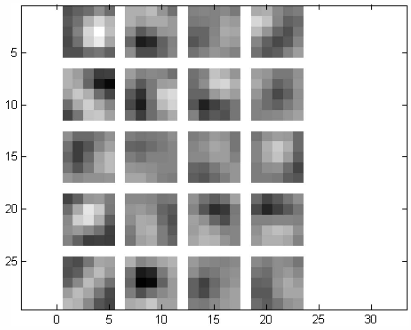


Fig. 8: Weights of convolutional layer in CNN1

In this paper, we test CNNs with a brain MRI apart from training set. Test image consists of 24725 patches. Accuracy rates of 4 CNNs are in Table 5. Accuracy rates of 2 ANNs are in Table 6.

Table 5: Accuracy rate of CNNs

Network	Average No. of Correct labels	Accuracy Rate
Proposed CNN	22458	90.83%
CNN1	22246	89.97%
CNN2	22299	90.18%
CNN3	21326	86.25%

Table 6: Training result on neural network for MRI

	Training error	Accuracy Rate
Neural Network 1	0.0124	76.68
Neural Network 2	0.0078	74.94

In the table, we can get that our proposed CNN outperformed the other 3 CNNs. CNN1 and CNN2 are similar on the result, but training time for CNN2 was shortened. CNN3 was the worst due to the structure and input patch size. Interior structural information of 13×13 patch is less than the 32×32. Proposed CNN performed the best due to the number of feature maps. Accuracy rate is limited to the training epoches from test results. Results of CNNs are far better than ANNs'.

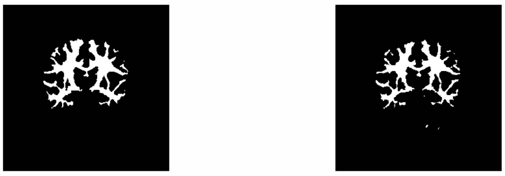
Besides accuracy rate, Dice Ratio (DR) is also used to measure segmentation accuracy. In this formula, A and B denote the manual annotation of segmentation labels and segmented by CNNs [18]. A larger DR value indicates a higher segmentation accuracy. The Dice ratio is defined as:

$$DR(A,B) = \frac{2 \mid A \bigcap B \mid}{\mid A \mid + \mid B \mid} \tag{5}$$

Dice Ratio of CNNs is showed in Table 7:

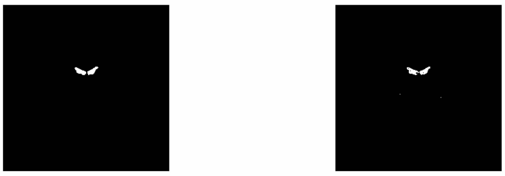
Table 7: Accuracy rate of CNNs

Network	Dice Ratio
Proposed CNN	95.19%
CNN1	94.12%
CNN2	94.83%
CNN3	92.62%



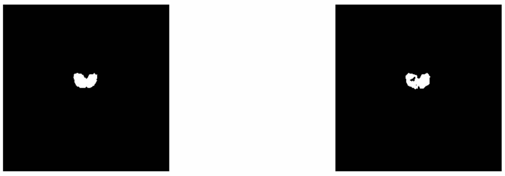
(1) Raw binary brain MRI (2) Binary segmented brain MRI

Fig. 9: Segmentation of Cerebral White Matter



(1) Raw binary brain MRI (2) Binary segmented brain MRI

Fig. 10: Segmentation of Lateral Ventricle



(1) Raw binary brain MRI (2) Binary segmented brain MRI

Fig. 11: Segmentation of Thalamus Proper

3.3 Segmentation of brain MRI results

Fig. 1, Fig. 9, Fig. 10 and Fig. 11 are raw images and segmentation results of proposed CNN.We segmented more than 90 percent region of brain MRI with convolutional neural networks. Accuracy rate of 90 percent outperformed most of traditional methods and machine learning methods. Meanwhile, we just used 100000 patches which extracted only in 4 brain MRI. Automatic segmentation method in this paper doesn't need big data and training process won't take too long. Trained CNNs can segment complex edge pixels successfully. But there are some pixels of misclassification.

4 Conclusion

In this paper, we proposed a new patch-based CNN for automatic segmentation of brain structure in MRI. It achieved better segmentation accuracy and provided a practical solution for the problem of insufficient training data with the introduction of patch-based CNN. Experiments demonstrated that the proposed method outperforms three other CNNs and two ANNs for brain MRI segmentation.

5 Acknowledgments

This work is partly supported by NSFC (No: 61375048).

References

[1] Shaun, Pacheco, et al. Brain MRI Segmentation, *Computational Surgery and Dual Training*, Springer US, 2010:45-73.

[2] Dan C. Cirean, et al. Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks, *Medical Image Computing & Computer-assisted Intervention: Miccai International Conference on Medical Image Computing & Computer-assisted Intervention*, 2013:411-418.

[3] Krizhevsky, Alex, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks, *Advances in Neural Information Processing Systems*, 25.2(2012):2012.

[4] Schlkopf, B., Platt, J., and Hofmann, T. Greedy layer-wise training of deep networks, *Advances in Neural Information Processing Systems*, 19(2007):153-160.

[5] Hinton G, et al. A fast learning algorithm for deep belief nets, *Neural Computation*, 2006, 18(7):1527-54.

[6] Srivastava, Nitish, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting, *Journal of Machine Learning Research*, 15.1(2014):1929-1958.

[7] Arnold L, Rebecchi S, Chevallier S, et al. An Introduction to Deep Learning, *ESANN*, 2011.

[8] Li R, Zhang W, Suk H I, et al. Deep learning based imaging data completion for improved brain disease diagnosis*Medical Image Computing and Computer-Assisted InterventionMICCAI* 2014. Springer International Publishing, 2014: 305-312.

[9] Dan, C Cirean, et al. Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images, *Advances in Neural Information Processing Systems*, 25(2012):2852–2860.

[10] Petersen, Kersten, et al. Breast Tissue Segmentation and Mammographic Risk Scoring Using Deep Learning, *Breast Imaging*, 8539(2014):88-94.

[11] Frazier, et al. Diagnostic and sex effects on limbic volumes in early-onset bipolar disorder and schizophrenia, *Schizophrenia Bulletin*, vol 34, no.1, p37-46, 2008

[12] Kennedy D N, Haselgrove C, Hodge S M, et al. CANDIShare: a resource for pediatric neuroimaging data. *Neuroinformatics*, 2012, 10(3): 319-322.

[13] Gonzlez, Alberto Martnez. Segmentation of brain MRI structures with deep machine learning, *Universitat Politcnica De Catalunya* (2012).

[14] Bouvrie J. Notes on convolutional neural networks. 2006.

[15] Leena, Silvoster M., and V. K. Govindan. Convolutional Neural Network Based Segmentation, *Computer Networks and Intelligent Computing. Springer Berlin Heidelberg*, 2011:190-197.

[16] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks*International Conference on Artificial Intelligence and Statistics*, 2011: 315-323.

[17] Yan, Pingkun, et al. Pixel-based machine learning in medical imaging. *International Journal of Biomedical Imaging*, 2012.4(2012):25 - 38.

[18] Zhang, Wenlu, et al. Deep Convolutional Neural Networks for Multi-Modality Isointense Infant Brain Image Segmentation, *Neuroimage*, 108(2015):214224.

[19] Dahl, J Vom, et al. Convolutional networks and applications in vision, *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on IEEE*, 2010:253-256.

[20] A. Vedaldi and K. Lenc. MatConvNet – Convolutional Neural Networks for MATLAB, *Proceeding of the ACM Int. Conf. on Multimedia*, 2015