DemNet: A Convolutional Neural Network for the detection of Alzheimer's Disease and Mild Cognitive Impairment

Conference Paper · November 2016

DOI: 10.1109/TENCON.2016.7848755

CITATIONS
69

READS
756

4 authors, including:

David Earl Hostallero
McGill University
9 PUBLICATIONS **126** CITATIONS

SEE PROFILE

DemNet: A Convolutional Neural Network for the Detection of Alzheimer's Disease and Mild Cognitive Impairment

Ciprian D. Billones, Jr., Olivia Jan Louville D. Demetria, David Earl D. Hostallero, and Prospero C. Naval, Jr.
Computer Vision & Machine Intelligence Group
Department of Computer Science
College of Engineering
University of the Philippines-Diliman
Email: pcnaval@dcs.upd.edu.ph

***Abstract*—The early diagnosis of Alzheimer's Disease (AD) and its prodromal form, Mild Cognitive Impairment (MCI), has been the subject of extensive research in recent years. Some recent studies have shown promising results in the diagnosis of AD and MCI using structural Magnetic Resonance Imaging (MRI) scans. In this paper, we propose the use of a Convolutional Neural Network (CNN) in the detection of AD and MCI. In particular, we modified the 16-layered VGGNet for the 3-way classification of AD, MCI and Healthy Controls (HC) on the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset achieving an overall accuracy of 91.85% and outperforming several classifiers from other studies.**

I. Introduction

Alzheimer's disease (AD) is a neurodegenerative disease that causes problems with memory and behavior and comprises 60%-80% of the cases of dementia. The most common symptom pattern of the disease begins with the gradual worsening of the ability to remember new information. A common practice in the diagnosis of AD involves the use of magnetic resonance imaging (MRI), a technique that makes use of magnetic fields and radio waves in order to create a 3D representation of the brain [14]. Through MRI data, radiologists identify the brain changes such as the appearance of tumor or evidence of a stroke. However, despite the efforts being made by different studies, the problem that there is currently no known cure for AD still persists.

Mild cognitive impairment (MCI) is a condition in which an individual's thinking ability shows some mild changes that can be easily noticed by the people who are close to the affected person. In the 2011 criteria and guidelines for diagnosis of AD, some cases of MCI are actually considered as early stages of AD. It is therefore an interest of researchers to detect the prodromal form of AD, MCI, before it progresses further and render symptomatic treatments ineffective [1].

As shown in recent studies, the use of convolutional neural networks for pattern recognition generally achieves competitive results in many classification tasks. In this paper, we present a modified version of the 16-layer VGGNet architecture that can be utilized together with the data preprocessing pipeline to solve the AD vs MCI vs HC classification problem. We report on the classification results of a 3-way CNN classifier and three independent CNN binary classifiers.

II. Related Works

Gupta et al. explored filter or base extraction using a sparse autoencoder in their study. They conducted this on two types of data: i) MRI data with dimensions $75\times95\times68$, normalized into an International Consortium for Brain Mapping template using Statistical Parametric Mapping, and ii) natural images that are of size 512×512, where 1000 patches of size 8×8 were sampled. After learning 100 bases that represent the lesions in MRI data, they proceeded with applying 2D convolutions on the MRI data with a kernel size of *8* and stride size of *1*. Feature activations were then obtained using the sigmoid function. In order to reduce the dimensionality, non-overlapping max pooling with a kernel size of *3* was applied. For the classifier, a fully connected neural network with a hidden layer size ranging from 800 to 1600 neurons was used [2].

Applying the same concept, Payan and Montana trained a sparse autoencoder to extract features and used the learned features for the convolutional layer of a CNN. The convolution operations were also followed by pooling operations similar to Gupta et al.'s neural network. This is then followed by a fully connected layer with 800 hidden units, and an softmax output layer with 3 outputs corresponding to the probabilities of the classes. The sparse autoencoder takes 100,000 $5\times5\times5$ patches sampled from 100 MRI scans of size $68\times95\times79$, and extracts 150 filters to be used in the 3D convolutional layer of their CNN, whose kernel size is *5* and stride size is *1*. The pooling layer after the convolutional layer has a kernel size of *5*. Payan and Montana also implemented a 2D convolutional neural network, with 150 filters of size 11×11 still extracted from a sparse autoencoder. The pooling operation in the 2D CNN used 10×10 pooling patches instead of $5\times5\times5$ [14].

Sarraf and Tofighi presented a solution to the binary classification of Alzheimer's Disease versus Healthy Controls from functional magnetic resonance imaging (fMRI) data. They have implemented the LeNet-5 architecture designed by Y.

978-1-5090-2597-8/16/$31.00

LeCun et al. [10] to learn features from the fMRI data and predict an output. Moreover, the LeNet model based on the CNN architecture from Caffe DIGITS 0.2 was used in their research [16]. The architecture they have adopted consisted of two convolutional layers with a max-pooling layer after each convolutional layer. The first convolutional layer has a kernel size of *5*, a stride of *1*, and 20 output neurons. The second convolutional layer has the same configuration as the first, except it has 50 output neurons. Both max-pooling layers have a kernel size of *2* and a stride of *2*. This architecture achieved a 96.86% accuracy in the binary classification of AD vs. HC.

III. Methods

A. Data Acquisition

Data used in the preparation of this article were obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (adni.loni.usc.edu). ADNI is an initiative that unites researchers that collect, validate, and utilize data such as MRI and PET images, genetics, cognitive tests, CSF and blood biomarkers as predictors for Alzheimer's Disease [11].

A total of 900 1.5T-weighted MRI scans in Neuroimaging Informatics Technology Initiative (NIfTI) file format were downloaded from the ADNI database. The scans were divided into three different classifications namely HC, MCI, and AD with each classification having an equal distribution of 300 MRI scans. These MRI scans were then further divided into our training set and testing set, having a 70-30 ratio respectively.

B. Pre-processing

To remove the unnecessary details of the MRI brain scans that might cause the network to learn unneeded features, cortical reconstruction and volumetric segmentation were performed with the Freesurfer image analysis suite, which is documented and freely available for download online (http://surfer.nmr.mgh.harvard.edu/). Specifically, the function *recon-all -autorecon1* was used, where only 5 out of 31 transformation processes were employed. The 5 processes are as follows: 1) Motion Correction and Conform; 2) Non-Uniform intensity normalization (NU); 3) Talairach transform computation; 4) Intensity Normalization 1; and 5) Skull Stripping.

After performing the cortical reconstruction processes, coronal image slices with size *256×256* in Portable Network Graphics (PNG) file format were extracted using Ubuntu's built-in medical data format conversion called *miconv*. From these slices, only the slices with indices 111 to 130 were used in the study, under the assumption that these slices cover the areas that have the important features for the classification task.

IV. The DemNet Convolutional Neural Network

After the data pre-processing step, we built a 2D convolutional network which we call DemNet, based on the VGGNet, that takes the slices as input. The core operations used in the 2D CNN are detailed below.

A. Convolutions

Convolution operations done on an image of size $h \times w$, with a kernel size of k, stride s, and padding p, produces an output of size $\frac{(h-k+2p)}{s+1} \times \frac{(w-k+2p)}{s+1}$. The kernels act as feature detectors, convolved with the image, thereby producing a set of convolved features [13].

In the neural network, the kernel size indicates the receptive field of a neuron, thus enforcing local connectivity of the neurons to the previous volume. In this study, all kernels used are of size *3×3* with a padding size of *1*, and stride of *1*. We also stack up to three convolutional layers before dimensionality reduction operations were done, so that more ReLU operations are incorporated, resulting in a more discriminative decision function [17]. The final configuration of the convolutional layers are shown in Table I of Section IV-G.

B. Rectified Linear Units (ReLU)

ReLUs, whose activation function is defined as $f(x) = max(0, x)$ were used instead of the sigmoid activation function or the hyperbolic tangent function. Training time using ReLUs is significantly faster than the equivalent hyperbolic tangent and sigmoid units [6].

All convolutional and fully-connected layers in this study are followed by ReLUs.

C. Pooling

Max-pooling is an aggregation operation that gets the maximum value in a region, specified by a kernel size k, an input with size $h \times w$, and stride s. The operation produces an output of size $\frac{(h-k)}{s+1} \times \frac{(w-k)}{s+1}$. When $s = k$, the regions operated on are non-overlapping [5].

Pooling does not only reduce the dimensions of the inputs, but it effectively summarizes the outputs of neighboring groups of inputs as well [6]. All pooling layers in the proposed network architecture are max non-overlapping, with kernel sizes of *2*.

D. Dropout

Dropout layers set the output of neurons in the hidden layers with a probability of r, called the dropout ratio, into 0. Neurons that are dropped out do not contribute to the forward pass and the backpropagation steps. In this way, the neural network samples a different architecture each forward-backpropagation step, but all of these still share weights. This technique is primarily used in an attempt to reduce test errors by avoiding overfitting[7].

In the architecture we propose, dropout layers were added every after pooling layers, and all dropout ratios were set to 25%, 40%, or 50%, as seen in Table I of Section IV-G.

E. Fully-Connected Layer

A fully-connected layer's neurons have full connections to all the activations in the layer before it. Three fully connected layers were used in the network at the end of the last pooling layer, with the number of output neurons as follows: *256-256-3* for the 3-way classification, and *256-256-2* for the binary classification.

F. Optimization

To decrease the training time of the network, the technique of transfer learning [18] was employed rather than using a random, Gaussian, or Xavier initialization. The proposed CNN was initialized using the weights of the VGGNet 16-layer network available in the Caffe Model Zoo. This model designed by Simonyan and Zisserman [17] contains good feature detectors for images included in the ImageNet dataset, which could also be utilized to detect edges from the images in our dataset. We used Stochastic Gradient Descent (SGD) as the optimization procedure [13].

G. Network Architecture

DemNet is a modified version of the 16-layer CNN made by the Oxford University Visual Geometry Group (VGG) for the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). It is composed of 13 convolutional layers and three fully connected layers. The convolution layers were divided into 5 batches by a max-pooling layer. The first two batches each contain 2 convolution layers while the remaining 3 batches has 3 convolution layers each. For this problem, we added dropout layers after each pooling layer to reduce overfitting. The first three dropout layers have a dropout ratio of *0.25* while the fourth and fifth have a dropout ratio of *0.4*. The number of outputs for the fully connected layers have also been modified to *256-256-3* from the original *4096-4096-1000*. The layer configurations along with their parameters are as described in Table I. The output is the number of neurons, dr is the dropout ratio, and k-s-p stands for kernel size, stride, and padding.

This convolutional neural network takes in an input of an MRI slice and crops it to a size of *224×224*. It then produces 3 outputs corresponding to the scores of each classification.

DemNet was trained and tested in both 3-way and binary classification. For the 3-way classification DemNet, only the accuracy was used to evaluate the classification. During testing, the output values of the 20 MRI slices were fed to a softmax function whose output is a vector of class scores. This produces a single vector of classification probabilities for each set of slices.

Training of the DemNet for the 3-way classification problem took a total of 11 hours in a system with NVIDIA GeForce GTX Titan X GPU.

We have achieved an overall accuracy of 91.85% for the 3-way classification, with AD being the easiest to classify, MCI being confused with both AD and HC, and HC being slightly confused with MCI. The confusion matrix of the 3-way classification is described in Table II, where the columns represent the instances in the predicted class while the rows represent the actual class.

The accuracy of DemNet is competitive with the previous works that used neural networks on MRI scans, achieving the best accuracy compared to the other CNNs created to solve the same problem. Table III shows the comparison of DemNet with other networks.

TABLE I: Proposed DemNet Architecture

Layer	Output	k-s-p	dr
conv1_1	64	3-1-1	
conv1_2	64	3-1-1	
pool1		2-2-0	
dropout1			0.25
conv2_1	128	3-1-1	
conv2_2	128	3-1-1	
pool2		2-2-1	
dropout2			0.25
conv3_1	256	3-1-1	
conv3_2	256	3-1-1	
conv3_3	256	3-1-1	
pool3		2-2-1	
dropout3			0.25
conv4_1	512	3-1-1	
conv4_2	512	3-1-1	
conv4_3	512	3-1-1	
pool4		2-2-1	
dropout4			0.40
conv5_1	512	3-1-1	
conv5_2	512	3-1-1	
conv5_3	512	3-1-1	
pool5		2-2-1	
dropout5		2-2-1	0.40
fc6_demnet	256		
dropout6		2-2-1	0.50
fc7_demnet	256		
dropout7		2-2-1	0.50
fc8_demnet	3		

TABLE II: 3-way Classification Confusion Matrix

Label\Output	HC	MCI	AD
HC	91.11%	7.78%	1.11%
MCI	6.67%	86.67%	6.67%
AD	0.00%	2.22%	97.78%

For the binary classification, the sensitivity and specificity were calculated in addition to the accuracy. DemNet has the best accuracy for AD vs HC, and AD vs MCI, even surpassing Payan and Montana's 3D convolution. The results and comparisons of the accuracy, sensitivity, and specificity are summarized in Tables IV - VI.

Accuracy for each slice index was also computed with the goal of examining the significance of different slice indices. A test of hypothesis was done using the *p-value* approach with α = 0.05 and the results indicate that the first and the last two slices (111, 129, and 130) have significantly lower accuracies than the average accuracy per slice index. This indicates that slices 112 to 128 contain enough features to classify the brain image, possibly making them sufficient representatives of the brain in this kind of classification problem. The accuracies of each slice is described in Figure 1. One possible reason goes back to the structure-based diagnosis of Alzheimer's disease [3], [4]. People with AD show significant shrinkage in their brain, specifically in certain biomarkers such as the hippocampus. The brain structure, which includes the hippocampus, is seen best in these slice indices.

V. Conclusion

In this study, we were primarily interested in exploring the application of Convolutional Neural Networks for the

TABLE III: Accuracy for 3-way Classification

3-way	Accuracy
DemNet	**91.85%**
Gupta et al. (Natural Bases)	85.00%
Gupta et al. (MRI Bases)	78.20%
Payan and Montana (2D Convolutions)	85.53%
Payan and Montana (3D Convolutions)	89.47%

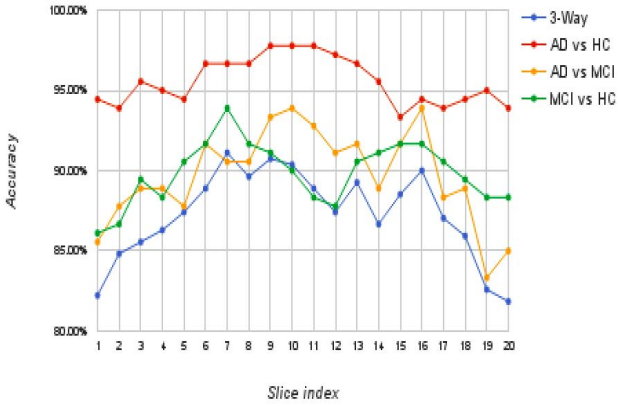

Fig. 1: Accuracy per Index

computer-aided diagnosis of Alzheimer's Disease and Mild Cognitive Impairment. We have explored the 16-layer VG-GNet and modified for this problem. Our proposed CNN architecture (DemNet) successfully classifies AD and MCI from HC on the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset with an accuracy of 91.85%, outperforming other CNN architectures on the ADNI dataset. Likewise, the binary classification achieved an accuracy of 98.33% for AD vs HC, 93.89% for AD vs MCI, and 91.67% MCI vs HC. The research also demonstrated that 17 coronal slices from the middle part of the brain is sufficient enough for classification. We have achieved these accuracies without segmentation of the gray matter, white matter, and cerebrospinal fluid, hinting that this classification problem may not be dependent on prior domain-knowledge and learned local features can be extracted to show variations between classifications.

TABLE IV: Accuracy

Classifiers	AD vs HC	AD vs MCI	MCI vs HC
DemNet	**98.33%**	**93.89%**	91.67%
Gupta et al. (Natural Bases)	94.74%	88.10%	86.35%
Gupta et al. (MRI Bases)	93.80%	86.30%	83.30%
Payan and Montana (2D Convolutions)	95.39%	82.24%	90.13%
Payan and Montana (3D Convolutions)	95.39%	86.84%	**92.13%**
Sarraf and Tofighi (fMRI)	96.86%		

TABLE V: Sensitivity

Classifiers	AD vs HC	AD vs MCI	MCI vs HC
DemNet	**98.89%**	**97.78%**	92.22%
Gupta et al. (Natural Bases)	92.24%	84.07%	**92.23%**
Gupta et al. (MRI Bases)	92.67%	84.55%	85.85%

TABLE VI: Specificity

Classifiers	AD vs HC	AD vs MCI	MCI vs HC
DemNet	**97.78%**	90.00%	**91.11%**
Gupta et al. (Natural Bases)	94.26%	**92.11%**	81.45%
Gupta et al. (MRI Bases)	94.92%	88.46%	80.99%

REFERENCES

[1] Alzheimer's Association, 2013 Alzheimer's Disease Facts and Figures, In *Alzheimer's & Dementia*, volume 9. Elsevier Inc., 2013.

[2] A. Gupta et al., Natural Image Bases to Represent Neuroimaging Data, In *Proc. of the 30th International Conf. on Machine Learning*, pages 987-994, Atlanta, GA, 2013.

[3] A. Bonner-Jackson etal., Verbal and non-verbal memory and hippocampal volumes in a memory clinic population

[4] LG. Apostolova Hippocampal atrophy and ventricular enlargement in normal aging, mild cognitive impairment (MCI), and Alzheimer Disease

[5] A. Karpathy et al., CS231n Convolutional Neural Networks for Visual Recognition.

[6] N. C. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A Simple Way to Prevent Neural Networks from Overfitting, In *Journal of Machine Learning Research*, vol. 15-1 pp 1929-1958, 2014.

[7] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, In *Advances in Neural Information Processing Systems 25*, pages 1097-1105, Stateline, NV, 2012.

[8] W. Zhu et al., Sensitivity, Specificity, Accuracy, Associated Confidence Interval and ROC Analysis with Practical SAS implementations, In *Northease SAS Users Group 2010 Proc.*, Baltimore, MD, 2010.

[9] Y. Jia et al., Caffe: Convolutional Architecture for Fast Feature Embedding, *arXiv preprint arXiv:1408.5093*, 2014.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-Based Learning Applied to Document Recognition, In *Proc. of the IEEE*, 86(11), 2278-2324.1998.

[11] Northern California Institute of Research and Education (UCSF). , Alzheimer's Disease Neuroimaging Initiative (ADNI). htttp://www.adni-info.org/. Last visit: January 15, 2016.

[12] LISA lab, Deep learning tutorials - deeplearning 0.1 documentation.

[13] A. Ng, Unsupervised Feature Learning and Deep Learning Tutorial, *CS294A Lecture*.

[14] A. Payan and G. Montana, Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks, In *Proc. of the International Conf. on Pattern Recognition Applications and Methods*, volume 1, pages 355-362, Lisbon, Portugal, 2015.

[15] M. Reuter et al., Within-Subject Template Estimation for Unbiased Longitudinal Image Analysis, *NeuroImage*, 2012, pp. 1402-1418.

[16] S. Sarraf and G. Tofighi, Classification of Alzheimer's Disease Using fMRI Data and Deep Learning Convolutional Neural Networks, *arXiv preprint arXiv:1603.08631*, 2016.

[17] K. Simonyan and A. Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, In *3rd International Conf. on Learning Representations*, San Diego, CA, 2015.

[18] J. Yosinski et al., How transferable are features in deep neural networks?, *https://arxiv.org/abs/1411.1792v1*, 2014.