CNN Assisted Parkinson's Disease (PD) Diagnosis Using MR Images

Luchen Liu
Dept. Computer Science
University of Alberta
Edmonton, Alberta, Canada
luchen@ualberta.ca

Yiting Dong
Dept. Computer Science
University of Alberta
Edmonton, Alberta, Canada
yiting5@ualberta.ca

***Abstract*—Various region of the brain has been researched and studied for locating bio-markers and diagnosing Parkinson's disease (PD). Previous Computer-Aided Diagnosis (CAD) studies of PD are based on analyzing clinical data by using PCA mapping and SVM classification. Instead of using linear mapping method to find features, we present an end-to-end CNN learning approach for finding the highly correlation region between Gray matter (GM)/ White matter (WM) with PD diagnosis by using preprocessed MR-Images from PPMI dataset. The method implemented in this paper is an end-to-end three-dimension neural network which is capable for learning features and classifying PD from HC with giving the GM and WM MR brain Images. A two-dimensional convolutional neural network (CNN) is also implement for visualizing and comparing the difference between different brain slices. Moreover, learning rate is optimized by testing on the three dimension CNN.**

***Index Terms*—Parkinson's Disease, MR Images, Convolutional Neural Network, Deep Learning Framework, Computer Aided Diagnosis**

I. Introduction

It has been widely recognized that the Parkinson's Disease (PD) is the second most common neuro-degenerative disease in the world. The daily life of more that one percent of elderly people over 65 years old are seriously affect by the Parkinson's Disease. [1] Although PD has been clinical defined and studied for decades, the mechanism of PD still remains unclear. [2] Therefore, the early diagnosis of PD by finding and utilizing effective bio-markers as well as developing effective PD identification methods are very essential.

For most of cognitive impairment and neuro-degenerative diseases, analyze brain structural changes by using medical imaging techniques have been proven to be helpful. [3] [4] Among all medical imaging methods, Magnetic Resonance (MR) imaging has shown its superior characteristic in brain structure analysis because of its high contrast and resolution within soft tissue. With the rapid development of Machine Learning and Deep Learning techniques in recent years, the ML-based and DL-based brain classification has been widely used in Computer Aid Diagnosis (CAD) of many common neuro-degenerative diseases, such as the Alzheimer's disease, Mild cognitive impairment, Huntington disease, and Parkinson's Disease etc. [2]

A. Previous Methods and Results

Various ML-based methods [1] [2] [5] and DL-based methods [6] [7] have been applied by researchers trying to figure out effective bio-markers and improving performance for Parkinson's Disease diagnostics. For ML-based methods, the general ideas are dividing the whole process into feature extraction and PD classification. Principal Component Analysis (PCA) is a method mainly used for high-correlated feature extraction; Support Vector Machine (SVM) with linear/ puk kernel, Multi-Layer Perceptron (MLP), Radial Basis Function Neural Network (RBFNN) etc. [1]are widely tested for the PD classification performance.

According to the Braak's neuroanatomical model of Parkinson's Disease [8], the Substantia Nigra (SN) region has been proven to have significant correlation with the progression of Parkinson's Disease as well as different levels of cognitive impairment and dementia. Therefore, within the image preprocessing stage, the Substantia Nigra (SN) region are widely used as the Region of Interest (ROI), and been further cropped out manually by doctors and neuroscientists for following feature extraction in ML-based PD diagnosis.

B. Project Goal and Strategy

However for our project, instead of using the known ROI - Substantia Nigra (SN) for doing the CAD, which has been widely proved effective, our purpose is trying to figure out other bio-markers/ sub-regions in White Matter (WM) and Gray Matter (GM) that are highly correlated with PD. The Substantia Nigra region is not included in White Matter or Gray Matter region, these three are separate sub-brain regions respectively.

Therefore, the project difficulties lie in: 1) the effective ROI or bio-markers that highly correlate with PD within WM and GM are not clearly known. 2) The labeling work for potential region or bio-markers needs help from experienced doctors or neuroscientists. To solve the second difficulty, we changed our strategy by using end-to-end training method [7] and no need for labeling the potential bio-markers. Besides, as proven by *Marlon O. et al.* that for classification, the highest accuracy performance could be achieved by using the end-to-end CNN training method comparing with other classifiers such as: SVM, MLP, LDA and K-NN etc. [9]

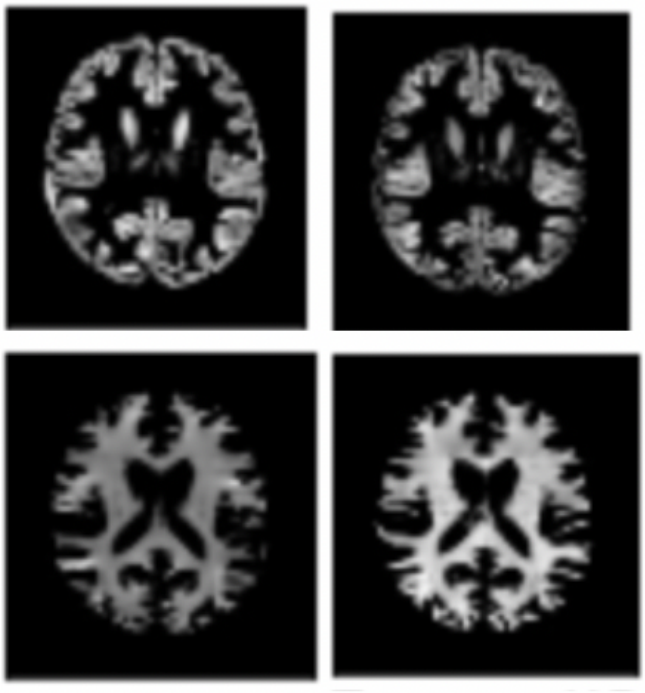

Fig. 1. Visualization of Gray Matter and White Matter. Upper-left: Gray Matter (GM) of PD group; Upper-right: Gray Matter (GM) of HC group; Lower-left: White Matter(WM) of PD group; Lower-right: White Matter(WM) of HC group

II. Methods and Materials

A. Data and Environment

1) Format and statistic of Data: The data used in this project is provided by University of Alberta, Multimedia Research Center(MRC) from the public Parkinson's Progression Markers Initiative (PPMI) dataset. [10] We use the three-dimensional gray-scale T1-weighted brain MRI images (.nii format) with the size of 121*145*121. Within all data, we got 358 3D-brain images for Parkinson's Disease patient (PD) group and 165 3D-brain images for Health Control (HC) group, which are in total 523 images for each category (WM and GM).

2) Preprocessing of Data: The Software used for doing the data preprocessing is MRICroGL and FSL. The original data were firstly preprocessed by stripping the skull. Followed by extracting White Matter (WM), Gray Matter (GM) and CerebroSpinal Fluid (CSF) separately. The illustration of WM and GM are shown in *Fig.1*.

3) Developing Environment: In this project, we chose to use *Python version 3.6* as the high-level programming language; the *OpenCV* programming function library for computer vision; and *Tensorflow* open-source software library for machine learning application.

III. Machine Learning Model

In this project, we divided our dataset as 75 percent for training and 25 percent for testing. Given dataset are separated into two classes, PD and HC based on a given *.CSV* format form. After the data preparation step, training dataset contain 365 subjects and 16 percent of the batch data will be selected as validation while training, testing dataset has 156 subjects.

In order to implement our project goals: *1) figure out the highly PD-correlated sub-regions 2) getting more information on how correlated WM and GM is with PD separately, as well as 3) better visualize the brain feature map,* we build up both 2D Convolution Neural Network model and a 3D Convolution Neural Network model.

For the 2D CNN model, the input images are 3-channel RGB images, and are re-sized to 128 by 128 by using *bilinear interpolation method* in openCV. For 3D CNN model, each input data is re-sized into 128*128*121 (128*128 stands for the size of 2D images on Coronal plane, and 121 stands for the number of brain slices along the sagittal direction).

Due to the large size of the 3D input images and limited computing power, we further divided our 3D brain images into several sub-volumes along z axis. Data are read into network batch by batch, and the batch size are set to 23.

A. The 2D CNN Model Structure

The origin 2D CNN model is a 2 by 3 followed by two fully connected layers image classification CNN model. The learning rate is defined as 1e-3, softmax is selected as the classifier at the last fully connected layer and *Adam Optimizer* is chosen for optimizing the model based on *Diederik P. Kingma and Jimmy Ba*'s paper [11].

$$t := t + 1 \tag{1}$$

$$lr_t := extLeraningRate\sqrt{(1 - beta_2^t)}/(1 - beta_1^t) \tag{2}$$

$$m_t := beta_1 * m_t - 1 + (1 - beta_1) * g \tag{3}$$

$$v_t := beta_2 * v_t - 1 + (1 - beta_2) * g * g \tag{4}$$

$$variable := variable - lr_t * m_t/(\sqrt{t} + \epsilon) \tag{5}$$

Batch size is set to 29, 2000 iterations are performed while training.

Batch normalization are added into the model in order to stabilize the learning and handling the over-fitting problem. The accuracy and loss comparison results of the train and validation can be find in the result of section 4-A *Fig.5*, with normalizing the output from each convolution layer, the training process can be significantly stabilized, and the over-fitting problem can also be reduced at the beginning of training. Therefore, we are using the 3 by 3 with 2 fully connected layers model as our final 2D model for further testing.

Specifically, the 2D CNN structure we built is shown in *Fig.2*. Layer L1, L4 and L5 are Convolution layers with 32,32 and 64 3 by 3 filters stride 1 ; layer L2, L5, and L8 are Batch Normalization layers; Layer L3, L6, and L9 are Max Pooling layers by using filters with size 2 by 2 and stride 1; L10 and L11 are Fully Connected layers.

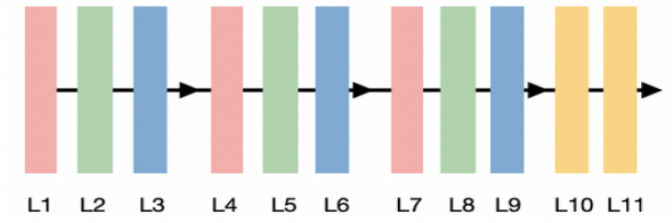

Fig. 2. Our 2D Convolution Neural Network structure

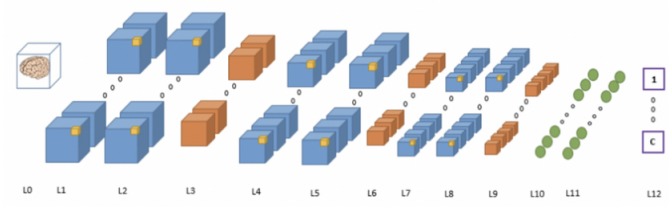

Fig. 3. The 3D Convolution Neural Network structure in the Stanford Reference Paper

B. The 3D CNN Model Structure

We got the inspiration of our 3D-CNN structure from another similar research from Stanford University [7] In which, *Soheil E. et al* proposed their 3D-CNN structure as shown in *Fig.3*. However their input images are the whole-brain 3D images, while our input images are White Matter and Grey Matter (sub-brain) 3D images. We modified our 3D-CNN structure base on the structure shown in *Fig.3* to better fit our experiment. Specifically, the 3D CNN structure we built is shown in *Fig.4*.

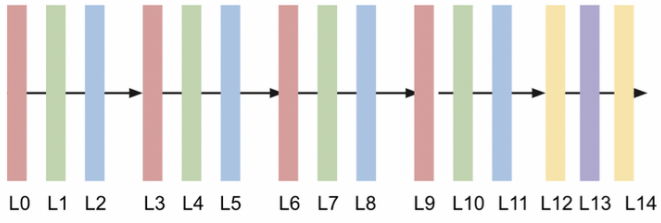

Fig. 4. Our 3D Convolution Neural Network structure

Input images into our network are 3D MR White Matter (WM) and Grey Matter (GM) images with the shape of 128*128*20. The layer L0, L3, L6 and L9 are Convolution layers with kernel size as 3*3*3, stride as one, padding method as 'same'. Layers L1, L4, L7 and L10 are Batch Normalization layers, these layers are added into the network for the purpose of accelerating the process of convergence as well as overcoming the over-fitting problem (Due to the small size of training dataset). Layers L2, L4, L8 and L11 are Max Pooling layers, among which, in L2, L4, L8, stride is set as 4*4*2, using the *'same' padding* method; and in layer L11, the stride is set as 2*2*2, using the *'Valid' padding* method. L12 and L14 are Fully Connected layers, and in between, L13, a dropout layer with 70 percent keep-probability is added in the training process also for the purpose of overcoming the over-fitting problem caused by our small training dataset size.

The activation method we used here in this model is the *ReLu function (tf.nn.relu)*, which looks for the max-value between input value with zero. For doing the final classification, we use the *SoftMax classifier* by utilizing the softmax cross-entropy (tf.nn.softmax) loss, where the y is the truth value adnd s is the softmax value.

$$s_i = e_i / \sum_i e^j \tag{6}$$

$$Loss = -\sum_j y_i ln s_i \tag{7}$$

Besides, we use the *Adam Optimizer* for the training optimization and by tuning the hyper-parameters, and we set the learning rate at 5e-4 according to results. We count a correct prediction when getting a 'True' value by using the (tf.equal) function to compare the predicted class with the true class and the accuracy is then calculated by taking the mean value.

Specifically in our case, the 3D model with a large-size 3D image as input is complex and involves many variables, which may lead to unstable learning if using unsuitable learning rates. Hence, we explored some learning rate that have previous been used in neurological studies to determine the best fit learning one for our model. The selected learning rates for the experiment are 5e-2, 1e-2, 5e-3, 1e-3, 5e-4, 1e-4 and 5e-5, we made comprehensive evaluation based on the performance, efficiency and accuracy of each learning rate. According to the results we got, with using 5e-4 as the LR, the network outperformed on balancing the efficiency, stability and accuracy of the model. In conclusion, we continued the study with 5e-4 as our default learning rate.

We loaded in our training images batch by batch and the batch size is set to 23 according to different batch size performance. Within each batch, we randomly separated all images into 84 percent for training and 16 percent for validation.

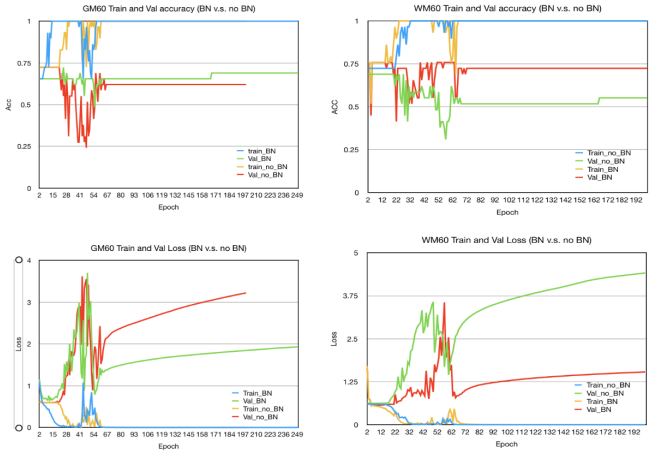

Fig. 5. Training and validation accuracy with and without Batch Normalization for 2D CNN

TABLE I
THE 2D COMPARISON RESULTS BETWEEN DIFFERENT SLICES OF GM AND WM .

Region	slice 60	slice 65
GM Training	1	1
GM Validation	0.6207	0.6207
WM Training	1	1
WM Validation	0.7162	0.6948

TABLE II
THE 3D COMPARISON RESULTS BETWEEN DIFFERENT SUB-REGION OF GM AND WM.

Region	slice[30:50]	slice[50:70]	slice[70:90]
GM Training	1	1	1
GM Validation	0.6250	0.6458	0.6667
WM Training	1	1	1
WM Validation	0.6087	0.7142	0.6754

IV. EXPERIMENT AND RESULTS

A. 2D implementation and result

For the 2D CNN training, we visulized the feature map as shown in *Fig.6*. On the basis of evaluating the middle slice from the brain data, we only achieve 62 and 71 percent of validation accuracy. We noticed that the result only depend on a single slice of data is not accurate enough because the information from the same slice of every subject's brain are various, we need to use CNN to explore more information from different slices. For getting a more accurate and confident result, at this point, we decided to explore different input image data to adjust our structure and improved result. We moved forward to feed different slice of the brain data to the CNN, and try learn different features from different slices. We compared the 60th and 65th slice of the brain for GM and WM data. However, the result shown in *Table 1* is still not clear enough to proof the correlation of GM/ WM and PD. In consideration of the inefficiency of evaluating the whole 121 slices of each subject's MRI data by using the 2D CNN, we designed a more comprehensive and efficient 3D CNN model to continue the study.

B. 3D implementation and result

According to the basic knowledge that our brain is roughly a ball-shape-like 3D volume and with larger volume in the middle region. Therefore, we could get larger 2D slice area if doing projection to all three radiological imaging planes (transverse, coronal and sagittal plane is shown in *Fig.7* and real MRI images are shown in *Fig.8*). Larger regions would contain more effective brain-structure information (non-zero value pixels/voxels from the image processing point of view) than smaller regions.

We took the coronal plane as analyzing plan in our experiments. The whole brain MRI image was re-sized to 128*128*121 along x, y and z axis respectively. As for the fact that if we use the whole 3D brain-volume images (128*128*128) as input to our fourteen layers 3D Convolution

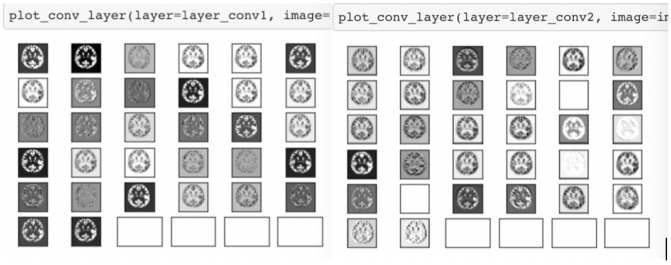

Fig. 6. 2D Feature map of GM for visualization

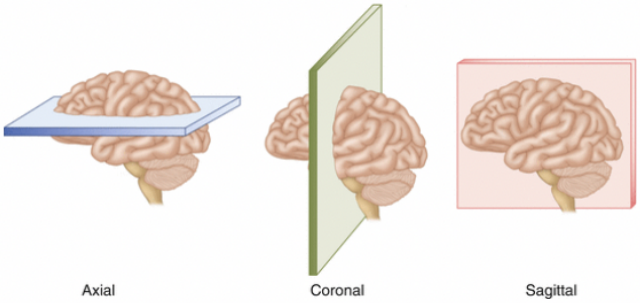

Fig. 7. Three radiology imaging orientations [12]

Neural Network, the computing parameters could be huge. Besides the computing power issue, such a strategy would also result in many other issues, for instance the difficult-parameter-tuning and severe over-fitting etc.

Since our purpose is to figure out the most correlated sub-volume in WM and GM respectively with PD, we changed our strategy to use smaller sub-volumes of 3D images (with 20 slices along z-axis) as input instead of whole brain volume (with 121 slices along z-axis). We separated the original images along sagittal direction into slice [0:30], slice [30:90] and slice [90:120]. For the reason that the medium 60 slices contain more effective information, we chose to analysis this sub-volume first and further separate them into slice [30:50], slice [50:70] and slice [70:90]. *Table 2* shows the experiment and GM/WM different sub-volume comparing results.

We added more supplementary data after the presentation, and got the final result as shown in Table2. As illustrated in the Table2, the accuracy for using White Matter images as input is slightly higher than using Gray Matter images as input, which is consistent with the results we got in ection 4-A.

Within WM, the best performance 0.7025 was got in the sub-volume from slice 50 to slice 70, later followed by the region from slice 70 to 90. Within GM, the best performance was got in the sub-volume from slice 70 to slice 90 and followed by the region from slice 50 to 70.

V. CONCLUSION

Both 2D and 3D CNN model are implemented and optimized end-to-end for investigating the correlation between GM/WM and PD through the study. The 2D model evaluates the information from a single slice of the brain input image, the

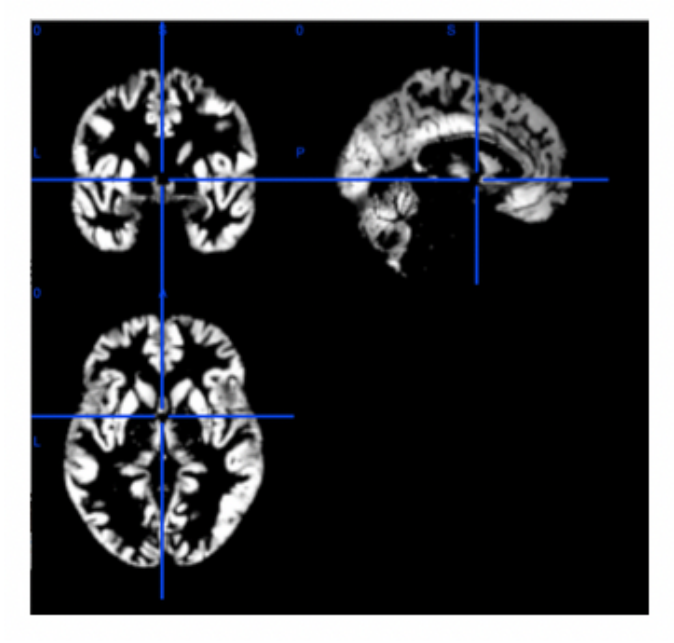

Fig. 8. Visualize the 3D Gray Matter whitin three planes: Coronal plane(left-up corner); Sagittal plane (right-up corner); Transverse plane(left-bottom corner)

model gives acceptable visualizing features results for offering a simple understanding to the learning process. Whereas the information of a 3D brain MRI is very complex and could various from slice to slice and subject to subject, the 3D model is introduced to study several different region of the brain data and produce a more comprehensive and detailed learning results.

According to our final result, the sub-region slice 50 to 70 in White Matter has the largest potential including bio-markers for PD diagnosis. Besides, generally speaking, White Matter gives better accuracy performance than Gray Matter, which the results we got in *section 4-B* by 3D model is consistent with the results we got in *Section 4-A* 2D model.

The preparation of the dataset involves image segmentation and reprocessing, different types of the image are also separated for different model. As optimizing our 3D model, we also determined an more adaptive learning rate for our case.

Moreover, in order to mitigate the over-fitting problem, batch normalization is added after each convolution layer into both 2d and 3d models, and dropout layer is concatenate in the fully connected layer of the 3d model as well. Even though batch normalization and dropout layer help with solving the over-fitting problem, because of the small data size, large image size and complex network, the over-fitting problem could still noticeable. We also find out the uneven number of data between PD and HC could also affect the learning of models, but since we are only provided with limited data and time, the data augmentation, features combining (age, gender,geographic locations [13] etc) could be conducted as future work.

ACKNOWLEDGMENT

The authors would like to thank the project instructor - Sara Soltaninejad and the course instructor - Lihang Ying as well as many other people in the MRC Lab for providing supportive guidance and instructions. The authors would also thank to the University of ALberta Multimedia Research Center for providing the server for GPU computation, we would not have our project done without the GPU server support.

REFERENCES

[1] S. Pazhanirajan and P. Dhanalakshmi, "Classification of parkinson's disease using mri images," *International Journal of Computer Science and Software Engineering*, vol. 5, no. 10, p. 233, 2016.

[2] B. Peng, S. Wang, Z. Zhou, Y. Liu, B. Tong, T. Zhang, and Y. Dai, "A multilevel-roi-features-based machine learning method for detection of morphometric biomarkers in parkinson's disease," *Neuroscience letters*, vol. 651, pp. 88–94, 2017.

[3] A. Sakalauskas, A. Lukoševičius, K. Laučkaitė, D. Jegelevičius, and S. Rutkauskas, "Automated segmentation of transcranial sonographic images in the diagnostics of parkinson's disease," *Ultrasonics*, vol. 53, no. 1, pp. 111–121, 2013.

[4] B. Rana, A. Juneja, M. Saxena, S. Gudwani, S. S. Kumaran, R. Agrawal, and M. Behari, "Regions-of-interest based automated diagnosis of parkinson's disease using t1-weighted mri," *Expert Systems with Applications*, vol. 42, no. 9, pp. 4506–4516, 2015.

[5] C. Salvatore, A. Cerasa, I. Castiglioni, F. Gallivanone, A. Augimeri, M. Lopez, G. Arabia, M. Morelli, M. Gilardi, and A. Quattrone, "Machine learning on brain mri data for differential diagnosis of parkinson's disease and progressive supranuclear palsy," *Journal of Neuroscience Methods*, vol. 222, pp. 230–237, 2014.

[6] H. Choi, S. Ha, H. J. Im, S. H. Paek, and D. S. Lee, "Refining diagnosis of parkinson's disease with deep learning-based interpretation of dopamine transporter imaging," *NeuroImage: Clinical*, vol. 16, pp. 586–594, 2017.

[7] S. Esmaeilzadeh, Y. Yang, and E. Adeli, "End-to-end parkinson disease diagnosis using brain mr-images by 3d-cnn," *arXiv preprint arXiv:1806.05233*, 2018.

[8] H. Braak, K. Del Tredici, U. Rüb, R. A. De Vos, E. N. J. Steur, and E. Braak, "Staging of brain pathology related to sporadic parkinson's disease," *Neurobiology of aging*, vol. 24, no. 2, pp. 197–211, 2003.

[9] M. Oliveira, H. Chatbri, S. Little, N. E. O'Connor, and A. Sutherland, "A comparison between end-to-end approaches and feature extraction based approaches for sign language recognition," 2017.

[10] K. Marek, D. Jennings, S. Lasch, A. Siderowf, C. Tanner, T. Simuni, C. Coffey, K. Kieburtz, E. Flagg, S. Chowdhury *et al.*, "The parkinson progression marker initiative (ppmi)," *Progress in neurobiology*, vol. 95, no. 4, pp. 629–635, 2011.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[12] H. A. Ahmad, H. J. Yu, and C. G. Miller, "Medical imaging modalities," in *Medical Imaging in Clinical Trials*. Springer, 2014, pp. 3–26.

[13] T. Pringsheim, N. Jette, A. Frolkis, and T. D. Steeves, "The prevalence of parkinson's disease: A systematic review and meta-analysis," *Movement disorders*, vol. 29, no. 13, pp. 1583–1590, 2014.