UNDERSTANDING 3D CONVOLUTIONAL NEURAL NETWORKS IN ALZHEIMER'S DISEASE CLASSIFICATION

by

JIAOJIAO WANG

(Under the Direction of Yi Hong)

ABSTRACT

The 3D Convolutional Neural Networks have achieved great success in many applications, including Alzheimer's Disease classification on medical image volumes. To increase their classification transparency and verify their prediction credibility, we uncover the 3D classification networks applied on 3D MNIST and OASIS-2, using two visualization techniques in deep learning, i.e., the Class Activation Mapping (CAM) and Layer-wise Relevance Propagation (LRP). We evaluate the performance of their resulting heatmaps in representing the relevance scores to the network's prediction from three perspectives: 1) visual interpretability, 2) quantitative measurement based on the Area Over the Perturbation Curve (AOPC), and 3) sanity check. The experimental comparison between CAM and LRP shows that CAM suffers the inconsistency between visual interpretability and heatmap quality, and LRP locates visually more meaningful regions for classification while could fail the sanity check.

INDEX WORDS: 3D CNN, Explanation, CAM, LRP, AOPC, Sanity Check

UNDERSTANDING 3D CONVOLUTIONAL NEURAL NETWORKS IN ALZHEIMER'S DISEASE CLASSIFICATION

by

JIAOJIAO WANG

BS, Shanghai University, China, 2013

MS, University of Missouri, 2015

MS, University of Georgia, 2020

A Thesis Submitted to the Graduate Faculty of The University of Georgia in Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

ATHENS, GEORGIA

2020

UNDERSTANDING 3D CONVOLUTIONAL NEURAL NETWORKS IN ALZHEIMER'S DISEASE CLASSIFICATION

by

JIAOJIAO WANG

Major Professor:	Yi Hong
Committee:	Sheng Li
	Shannon Quinn

Electronic Version Approved:

Ron Walcott
Interim Dean of the Graduate School
The University of Georgia
August 2020

ACKNOWLEDGEMENTS

First of all, I would like to express my deep and sincere gratitude to my dear advisor Dr. Yi Hong for giving me the opportunity to do research with her. Her vision, deep understanding and broad knowledge, sincere interest in medical image analysis research have inspired me so much since the very first day I knew her. The guidance she provided at each step is invaluable to me and her passions about research has set an excellent example for me. She has patiently and clearly taught me the methodology to conduct research. It was absolutely a great honor to work and study under her guidance. I am also very grateful for Dr. Sheng Li and Dr. Shannon Quin for serving in my committee in spite of their very busy schedules. Their kind and encouraging comments, considerate and patient attitudes throughout the whole time means very much to me. Without their input and guidance, it is impossible for me to complete this thesis.

I also would like to give special thanks to my dear lab mate Ankita Prashant Joshi. She kindly and patiently helped me with the 3D MNIST registration and is always ready to help as much as she can. My gratitude also extends to my other lab mates Raunak Dey and Rutu Gandhi, who gave many encouragements and great suggestions throughout my research.

Last but not least, I am so grateful for the countless prayers, uplifting encouragements and continuing support from my fiancé, my families, my roommate and all of my dear friends that I can’t mention one by one here.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

CHAPTER 1

Introduction and Background

Chapter 1.1 Thesis Overview

Alzheimer's disease (AD), the most common form of dementia, usually affects people over the age of 65. Once it starts to develop, the patient's cognitive and functional performances will be seriously impacted [1]. According to Alzheimer's Association, 1 in 3 seniors dies with Alzheimer's or another dementia and it is the 6th leading cause of death in the United States. Between 2000 and 2018, deaths from heart disease have decreased 7.8% while deaths from Alzheimer's have increase 146%. But 50% of primary care physicians believe that the medical profession is not ready for the growing number of people with Alzheimer's or other dementias. Thus, it’s not practical to always expect enough well-trained medical staff to scrutinize every neuroimaging output in clinical routine and automated image analysis using Machine Learning models are desperately needed.

Nowadays Convolutional Neural Networks (CNN) have achieved great success in many fields, including the automatic AD classification tasks. But is mere high prediction accuracy convincing enough for the medical staff to trust the diagnosis from a computer program? As we know, CNN is still deemed as a black box and we don’t fully understand how it makes a certain decision yet. For 3D CNN in AD classification, the trust level is even lower considering the relatively small amount of training samples and large number

of the parameters in the model. Without making this black box more transparent, it's hard for the neural network to play a role in clinical routine in spite of the high testing accuracy.

Thus, in this thesis, I am devoted to work on the explanation of 3D CNN's decision in the context of AD classification. In terms of the explanation, the most intuitive and widely accepted way is to generate heatmaps for a certain prediction from the model. It works by assigning each pixel a score representing how relevant it is to the final decision, and then visualizing it as an image in the same size as the input image. Two heatmap methods have been explored to explain 3D CNN's decision for AD in OASIS-2 dataset. The first one is Class Activation Mapping [4], where the heatmap is generated by simply calculating the weighted sum of feature maps and then up-sampling to the original image size. To understand this method deeper, we also studied the correlation between the heatmaps and model complexity besides simply generating heatmaps. The second method is Layer-wise Relevance Propagation (LRP) [6]. It works by propagating the predicted output value back layer by layer till the input layer is reached, during which the relevance score is distributed depending on contribution percentage during the forward propagation. Among the various LRP rule variations designed for different purposed, we explored three composite rule sets.

To evaluate the heatmaps from different methods objectively, we take three approaches. The first one is to visually interpret them and identify relevant features for classification. The second one is to use a numeric metric that helps us quantify the heatmap quality. The method is called Area over the Perturbation Curve (AOPC) [7], which is based on the assumption that the pixels that are assigned high relevance scores should be the most likely

to destroy the prediction value if they are perturbed. In AOPC, several implementation variations are explored, such as the stride size, different strategies to divide and sort cubes. The last approach we take to evaluate heatmap is Sanity Check, which is to further verify the soundness of the heatmap. The intuition behind this check is that if the parameters of the original model are randomized, the produced saliency map should change. The parameters randomization is performed in cascading fashion from top to bottom layer, and then generate corresponding heatmaps to see if any change.

To verify our observations on OSAIS-2 dataset, all experiments are also conducted on the 3D MNIST dataset for an easy interpretation. We observed that in CAM the more layers we have in a model, the higher the prediction accuracy and the heatmap quality, but the lower visual interpretability. It shows us that CAM suffers the inconsistency between visual interpretability and heatmap quality due to its limited ability to transform the high-level abstract features back to the visual space in a complex model. We also found that LRP has a higher potential to identify specific features at an individual level in explaining the AD classification task than CAM. The same pattern is also observed in the 3D MNIST dataset where LRP highlights the features specific to a digit while CAM either uses the overall contour or highlights most regions. But that doesn't mean the LRP method is always perfect. For some LRP rules where only positive contribution is used in the back propagation, it would generate a multiplication chain of positive matrices which might converge at a certain layer. This causes LRP to fail the Sanity Check in OASIS-2 dataset, while CAM survived the check in both datasets. But this is not an issue for all heatmaps generated by LRP. For instance, the heatmaps generated by LRP in 3D MNIST dataset

survived the Sanity Check. It is because the much smaller input size slowed down the convergence speed.

From what we observed in heatmaps generated by LRP where specific features can be highlighted for individuals, 3D CNN seems to be trustworthy and has great potential to be used in the clinical routine in the future. However, the defects we observed about both heatmap methods make us less confident about the aforementioned statement. To explain 3D CNN's decision with more confidence, better heatmap methods need to be designed in the future. An ideal heatmap should be able to map the high-level abstract features, which the complex 3D CNN relies on to make decision, back to visual space without any loss or distortion. At the same time, the visual interpretability should correlate positively with the heatmap quality measurement, and it should also survive the very intuitive Sanity Check.

Chapter 1.2 Alzheimer's Disease

As we mentioned above, Alzheimer's disease (AD) is terribly affecting the seniors around the world. Sadly, so far there is no treatment that could stop or reverse its progression. But if AD could be diagnosed at an early stage, we could help the patients and families get prepared or even slow down its progression. Thus, preclinical diagnosis is needed for the prevention to battle against AD. But is it possible to detect AD before it could be diagnosed clinically? As we know that AD is related to brain lesions. Studies have shown that some of these lesions begin to form 20 to 30 years before the disease becomes clinically evident, which gives us the hope for early detection. However, it is a very challenging task to do the measurement especially for the purpose of early diagnosis where the brain lesion areas

are still small. Fortunately, several modern neuroimaging modalities already show promising results as early diagnostic tools for AD. Among which, Magnetic resonance imaging (MRI) is one of the most effective tools for structural assessment. Generally speaking, MRI is a medical imaging technique to form pictures of the anatomy of the body by using the strong magnetic fields, magnetic field gradients, and radio waves. In the differential diagnosis of AD from other type of dementias, MR imaging plays an important and routine role. In this thesis, this is also the modality of the brain imaging data we studied.

Generally speaking, there are many factors that could contribute to AD. Thus, which one is the real cause? In one study, the researchers recruited 45 healthy elders who are greater than 60 years old and conducted a 6 years longitudinal MRI imaging study of normal aging [1]. In the final observation, these subjects were separated into two groups who did and did not show objective evidence of cognitive decline. After many analyses were performed on these two groups, they found medial temporal lobe atrophy rate was the only significant predictor of decline in the normal subjects, while other factors such as age, sex, APOE genotype, education level didn't play significant roles. To identify which brain regions are related to AD more specifically, another study is conducted based on the concept of Mild Cognitive Impairment (MCI), which is commonly accepted as a transitional stage between normal aging and AD and can represents early stage AD [2]. In this study, they included 112 normal elderly individuals, 226 MCI and 96 AD subjects to test if constant and accelerated hippocampal loss can be detected in AD [3]. Over short period of time, the MCI and AD groups showed hippocampal volume loss over 6 months and accelerated loss over 1 year, whereas in the normal group hippocampal loss was detected over 1 year with

no indication of acceleration. From this study, they concluded that hippocampus is closely related to AD. Besides this study, there are also others that revealed more AD related brain regions. In summary, it is widely accepted that the shrinkage of hippocampus and cortex, and enlarged ventricles are the commonly accepted Regions of Interest (ROI) for AD.

Chapter 1.3 Machine Learning in Alzheimer's Disease Classification

With various modalities of neuroimaging data available, such as MRI, PET etc., how to use them in AD research? The original way is to have well trained medical experts to look at them and make judgement, and it is still the most trustworthy way. It takes many resources and time to train a radiologist, thus the workforce in this field is really limited comparing to the workload that need to be done. Considering the gap between the demand and supply of radiologists, it is not practical to have them scrutinize every image in the search for anomalies. With automated image analysis using Machine Learning models, the burdens for radiologists could be eased. It would also lower the cost for patients, speed up the overall healthcare flow and even has the potential to improve the diagnosis accuracy.

In the AD research field, many problems have been studied, such as the classification task that includes ternary classification between AD, MCI and NC, and binary classification between AD/NC, AD/MCI and MCI/NC. In this thesis, the classification task we focused on in explanation is AD/NC. Various Machine Learning (ML) methods have been applied to this field. Among the traditional ML methods, one research is to use Support Vector Machine (SVM) to do classification by generating eigenbrain [4]. They started by selecting key slices from 3D volumetric data using maximum inter-class variance. Then they

generated an eigenbrain set for each subject and obtained the most important eigenbrain by Welch's t-test. Finally, kernel support-vector-machines with different kernels were used to make predictions of AD subjects. The experiment showed that the proposed method can predict AD subjects with a competitive performance. The best accuracy achieved was 91.47% for linear kernel and 92.36% for polynomial kernel in OASIS-1 dataset.

Even though decent accuracy can be achieved in the proposed method, it requires manually designed features, which is the disadvantage of the traditional ML methods in general. For Convolutional Neural Networks (CNN), this is not a problem anymore since it allows for end-to-end training. Given the input and output, the CNN model is able to minimize human efforts and extract features automatically. At the same time, by not specifying which patterns to look for, it also gives us potential to identify new region of interest (ROI) for AD that has not been discovered before. Among the deep CNN methods, I want to mention this study, where the MRI and Positron Emission Tomography (PET) images from Alzheimer's Disease Neuroimaging Initiative (ADNI) database are used [5]. In the training process, they uniformly divided original input into patches to reduce computation cost. Because training a deep CNN model for the whole brain image requires high computation cost, especially for the high-resolution images. To keep the correlation information between MRI and PET images, they combine the multi-modality images at the same position. Then the features associated with both modalities are passed to higher level to make prediction. In this model, 93.26% accuracy was achieved in AD vs NC classification.

Transfer Learning is a ML method where a model trained for one task is reused as the starting point for another model on the second task. Usually relatively large number of labeled examples are available in the first source domain and that's why generic features can be extracted and transferred to new domains. As we know, MCI is the early stage of AD and has high change to convert to AD. Therefore, effective prediction of such conversion from MCI to AD is of great importance for early diagnosis of AD and also for evaluating AD risk. In this paper, they propose a transfer learning method that jointly utilizes samples from another domain as well as unlabeled samples to boost the performance of the MCI conversion prediction [6]. The source domain is samples of AD and Normal Control (NC). The target domain is MCI Converter (MCI-C) and MCI Non-Converter (MCI-NC). Experimental results on ADNI database validate the effectiveness of proposed method by significantly improving the classification accuracy of 80.1% for MCI conversion prediction.

Another interesting deep learning method Generative Adversarial Networks (GAN) has also been applied in the AD early detection field. In this research, they used Cycle GAN to generate missing PET images first and then use them for classification [7]. The reasoning behind this method is that multi-modality is more powerful and informative in the automatic detection. It is quite intuitive that different modalities carry different properties and they provide complementary information. But in practice, it is inevitable to have missing data. Thus, they used the existing MRI image to generate the missing PET images by using the underlying relevance between them. Then both MRI and PET images are used in the model training and it achieved 92.5% accuracy in the AD vs NC classification.

CHAPTER 2

Visualization and Evaluation Methods for Deep Neural Networks

2.1 Class Activation Mapping (CAM)

In Chapter 1, we talked about how the black box like 3D CNN models have the transparency issue. Due to the nonlinearities in the architecture, it is hard to grasp what features cause the neural network to arrive at a particular decision. To explain its decision, the most intuitive approach will be to generate corresponding heatmaps, where the pixel value represents the relevance score for how important that pixel is to the final prediction.

Class Activation Mapping (CAM) is the first heatmap method we used in this thesis. It was initially proposed in [9] and it shed light on the uniqueness of global average pooling (GAP) layer. Studies have shown that the convolutional layers have the ability to behave as object detectors despite no supervision on the location of the object was provided [10]. However, this remarkable ability is lost when fully connected layers are used for classification. Therefore, they avoid Dense layers in CAM except the last layer for the sake of classification. Instead, they used GAP layer right before the last Dense layer to make prediction and show how it explicitly enables the CNN to have remarkable localization ability despite being trained on image-level labels. Despite its apparent simplicity, they achieved 37.1% top-5 error for object localization on ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2014 without training on any bounding box annotation. They also demonstrated a variety of experiments that by using this method the network is

able to localize the discriminative image regions despite just being trained for solving classification tasks.

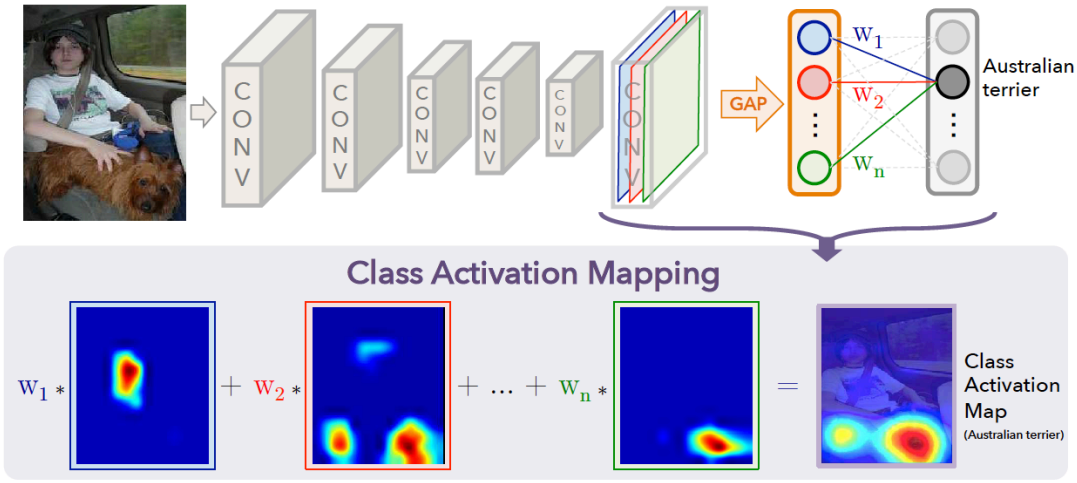

Figure 1: CAM Illustration [9]

The process for generating the Class Activation Map (CAM) using GAP is actually straightforward. As the name implies, a CAM generated for a specific label indicates the discriminative image regions used by the model to identify that label. The CAM is obtained by performing GAP on the convolutional feature maps right before the final output layer, which is the fully connected layer using softmax in the case of classification. In Figure 1, the overall process of CAM is depicted very clearly. To identify which regions are important for the Australian terrier prediction, weights of the output layer (W1, W2, … Wn) are projected back on to the convolutional feature maps right before GAP. Since GAP outputs the spatial average of the feature map of each unit at the last convolutional layer, a weighted sum of these values is used to generate the final output. Similarly, a weighted sum of the feature maps of the last convolutional layer are computed to be the class activation maps. Thus, the class activation map is simply a weighted linear sum of the presence of these visual patterns at different spatial locations. After we obtain the CAM

for the label Australian terrier, we need to simply up-sample the class activation map to the original input image size, then the image regions that are most relevant can be identified.

2.2 Layer-wise Relevance Propagation (LRP)

Layer-wise Relevance Propagation (LRP) is the second heatmap method we chose. It operates by propagating the final prediction value f(x) backwards layer by layer after the model finish training [11]. It starts at the last Dense layer where prediction is made and propagate all the way till the first input layer. Then the heatmap is generated using the relevance values that the first layer has received. The overall propagation process is illustrated clearly in Figure 2.

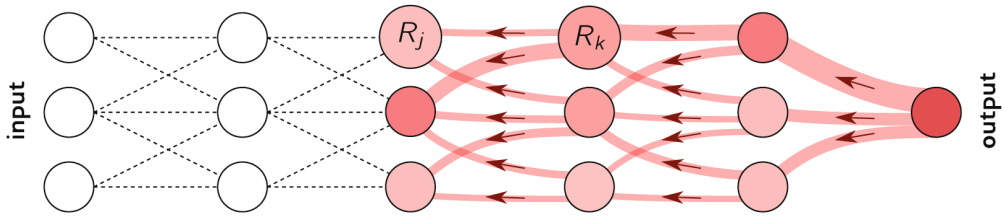

Figure 2: LRP Illustration [11]

To thoroughly understand how this back propagation works on the neuron level, we can start with Formula 1. Let's say neuron k is in layer L and neuron j is in layer L+1, which means neuron j and neuron k are at two consecutive layers in the neural network. Assume the relevance score has been propagated all the way to layer L+1 from the final output node, and R_k represents the relevance score neuron k received. Now the next step is how to propagate relevance score to neurons at layer L, such as neuron j? The answer is summarized in this formula. To begin with, the quantity Z_{jk} models the extent to which neuron j has contributed to make neuron k relevant in the final prediction. The denominator sums up all of the contribution neuron k received from the previous layer during the

forward propagation. Therefore, Z_{jk} divided by the denominator tells us how much neuron j contributions to neuron k to make it relevant in the final prediction in terms of percentage. Based on this measurement, the relevance score R_k flows back to neuron j proportionally. In the end, adding up the relevance scores propagated to neuron j from all the neurons in layer L+1, we will get the final relevance score R_j for neuron j. Using the same principle, we can calculate the relevance score for all neurons in layer L+1. Then Applying this rule recursively to the neurons at next layer till the input layer is reached, we will get the heatmap for a specific prediction.

$$R_j = \sum_k \frac{z_{jk}}{\sum_j z_{jk}} R_k \qquad (1)\ [11]$$

So far, we just looked at the very basic propagation rule, and there are many carefully designed variations for different purposes. Here, we will introduce four LRP rules that are used in the thesis, and they are LRP-0, LRP-ε, LRP-γ, LRP-αβ. The rule we just introduced above is the most basic one and it is called LRP-0. It is to simply redistribute relevance score to each neuron in proportion to the activation contribution. LRP-0 picks many local artifacts of the function. Therefore, the explanation is overly complex and does not focus sufficiently on the actual feature in the input. That's why more robust propagation rules are needed. LRP-ε is to add a small positive constant in the denominator, so that small noisy elements could be filtered out. As ε becomes larger, only the most salient explanation factors could survive the absorption. This typically leads to explanations that are sparser in terms of input features and less noisy. But if ε is set to be too big, the heatmap will become too sparse to be easily understood. Therefore, ε is a hyper-parameter we need to tune to find the best explanation. LRP-γ is to add a parameter γ to adjust how much positive

weights are favored. LRP-αβ works by adding two parameters α and β to adjust how much positive and negative contributions matters in the back propagation, and usually $\alpha = \beta + 1$. As we can see, when γ or α increase, negative contributions start to be suppressed and even disappear in the end. By treating positive and negative contribution in an asymmetric manner, the explanations become more stable. Usually the heatmaps generated by LRP-γ or LRP-αβ are easier to understand because features are more densely highlighted, but it might pick unrelated concepts which makes the heatmap not faithful to the label it is trying to explain.

Given the advantages and disadvantages of different methods, we can see why composite LRP rules is needed. For the lower layers in a deep neural network, which are the layers close to the input layer, LRP-γ and LRP-αβ are more suitable since they tend to spread relevance uniformly to the whole feature rather than capturing the contribution of every individual pixel. This makes the explanation more understandable for a human. For the middle layers, it has more disentangled representation. The stacking of many layers introduces spurious variations. LRP-ε filters out these spurious variations and retains only the most salient explanation factors. For the upper layers which are close to the final output, many concepts forming the different classes are entangled. Hence, a simple propagation rule which will be insensitive to these entanglements are needed, which are LRP-0. Using composite LRP rules, we can overcome the disadvantages of the single approach and the features can be identified and highlighted faithfully and clearly. In Table 1, you can find a comprehensive summary of the LRP rules and their proper usage in the model layer locations, which provided by the LRP overview paper [11].

Table 1: LRP Rules Summary [11]

Name	Formula	Usage
LRP-0	$R_j = \sum_k \frac{a_j w_{jk}}{\sum_{0,j} a_j w_{jk}} R_k$	Upper layers
LRP-ε	$R_j = \sum_k \frac{a_j w_{jk}}{\varepsilon + \sum_{0,j} a_j w_{jk}} R_k$	Middle layers
LRP-γ	$R_j = \sum_k \frac{a_j \left(w_{jk} + \gamma w_{jk}^+\right)}{\sum_{0,j} a_j \left(w_{jk} + \gamma w_{jk}^+\right)} R_k$	Lower layers
LRP-αβ	$R_j = \sum_k \left(\alpha \frac{\left(a_j w_{jk}\right)^+}{\sum_{0,j} \left(a_j w_{jk}\right)^+} - \beta \frac{\left(a_j w_{jk}\right)^-}{\sum_{0,j} \left(a_j w_{jk}\right)^-} \right) R_k$	Lower layers
z^β–rule	$R_i = \sum_j \frac{x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-}{\sum_i x_i w_{ij} - l_i w_{ij}^+ - h_i w_{ij}^-} R_j$	First layer

In practice, there are special layers in the neural networks that have no weights and biases associated, such as pooling layer, batch normalization and the input layer. They couldn't be handled by rules mentioned above and need to be addressed separately during the back propagation. For the max pooling layers, it can be handled either by winner-take-all redistribution scheme, or by using the same rules as for the sum-pooling case. As we know, batch normalization layers are commonly used to mitigate the problem of vanishing or exploding gradient. It helps the training process converge and improve the prediction accuracy. As reported in the paper [11], these layers can be absorbed by the adjacent linear layer without changing the function, which means applying identity rule for batch normalization layer could be enough in some cases. Input layer is different from intermediate layers as they do not receive ReLU activations as input but pixel values. Therefore, they need to be handled by LRP-Z^β rule, which you can see in Figure 3.

2.3 Area over Perturbation Curve (AOPC)

So far, we have looked at two widely used heatmap methods: CAM and LRP. By generating heatmaps to quantify the importance of individual pixels, we can get visual explanations for corresponding result. But how do we evaluate a heatmap? As human, we can assess the heatmap quality intuitively, by matching the heatmap with what we know is important for certain label. When we have small number of heatmaps for relatively simple images, this might work. But given many heatmaps generated by various methods, to rank them will takes enormous amount of efforts using this kind of individual manual evaluation. Additionally, human evaluations are always subjective, and it is very likely that different people will attribute different weights to the same feature when they evaluate a heatmap. Therefore, we need an objective method to quantify the heatmap quality.

A method called Area over Perturbation Curve is proposed for such purpose [13]. It is based on a very intuitive thought that comparing to the pixels with low relevance score, the high relevance score pixels should make the prediction value f(x) drop faster if they are perturbed. To test this expected behavior, they progressively remove information from the original input image x and measure how the final output value f(x) change. This process is referenced as region random perturbation. The very first step before any perturbations can be applied is to divide the original input image into patches. In the 3D cases, each patch will be a cube. There are different ways to do the division, and we experimented two in this thesis. The first way is to uniformly divide the image using a predefined grid. There is no overlapping between cubes, and they are right next to each other in the spatial domain. For instance, for the 3D MNIST data where input image is in the size of 32 x 32 x 32, if

we set the stride to be 4, then we will have 8 x 8 x 8 = 512 cubes, which means there are 512 cubes in total we can apply perturbation. The second way is to allow overlapping between cubes, which we call greedy search method. At each perturbation step, we search for a new cube that gives the maximum value using the Heatmapping Function h_p = H(x, f, r p). In this method, we can always focus on the most relevant cube.

After we have a list of divided regions, the next step will be to assign each region a priority for perturbation. Basically, we need to sort the original list into an ordered sequence, which can be represented as O = (r_1, r_2, . . . , r_L). This sorting process is guided by a heatmapping function h_p = H(x, f, r p), which assigns each cube a priority value based on the heatmap. After the sorting, all indices of the sequence O should satisfy the following property in Formula 2. It means the relevance of each cube is coded in the ordering of the sequence O. Locations in the image that are most relevant for the class encoded by the classier function f will be found at the beginning of the sequence O, and regions that are mostly irrelevant will be positioned at the end of the sequence.

$$(i < j) \Leftrightarrow (H(x, f, r_i) > H(x, f, r_j)) \qquad (2)\ [13]$$

After the input image is divided and the cubes are sorted, we are ready to apply the perturbation. In this thesis, a uniform distribution is used for the perturbation. The overall process can be summarized using the Formulas 3. Here are the meanings of the symbols so it's easier to grasp the essence of the perturbation process. L represents the number of perturbation steps in total, and k stands for the current step number. MoRF is short for Most Relevant First sequence, which is the ordered sequence O we introduced in the previous sections. X stands for the original input image, and $X_{MoRF}^{(k)}$ stands for the input image after

k_{th} perturbation. Thus, $X_{MoRF}^{(0)}$ is also the original input image X. As we can see in Formula 3, after applying the perturbation function g on region r_k and the previously perturbed image $X_{MoRF}^{(k-1)}$, we get the perturbed image at current step, which is $X_{MoRF}^{(k)}$. We can see that the perturbation effect is accumulated at each step. Apparently, at the last step L, all of the L regions in the images have been changed. L is a parameter that should be decided based on the computational resources we have at hand and also the size of the image. It is closely related to perturbation coverage percentage.

$$x_{MoRF}^{(0)} = x$$
$$\forall\, 1 \le k \le L: x_{MoRF}^{(k)} = g(x_{MoRF}^{(k-1)}, r_k) \quad (3)\ [13]$$

After having obtained the image at each perturbation step $X_{MoRF}^{(k)}$, we are ready to calculate the final Area over Perturbation Curve (AOPC), which is well defined in Formula 4. We start by calculating the final prediction score $f(X_{MoRF}^{(k)})$ for image perturbed at each step by applying the prediction function. Then calculate the difference with the original prediction value. Then we sum up all of the changes at each step and normalize it using the number of total perturbation steps. As we can imagine, if we only plot the AOPC curve for one image, it might look jagged. At the same time, given hundreds of testing images, how do we decide which image to choose? Obviously, cherry picking can't give us a reliable curve, since we can't trust it to reflect the overall performance of the model. Therefore, usually more than one images will be chose to be calculated, then their corresponding AOPC curves will be averaged as the final curve for the model. So that we can get a reliable and stable curve. In this thesis, we used 100 images in the 3D MNIST dataset and 30 images in the MRI dataset after balancing the computational resources and reliable performance. In the AOPC formula, the average over all images in the dataset is denoted by p(x).

$$AOPC = \frac{1}{L+1}\left(\sum_{k=0}^{L} f\left(x_{MoRF}^{(0)}\right) - f\left(x_{MoRF}^{(k)}\right)\right)_{p(x)} \quad (4)\ [13]$$

2.4 Sanity Check for Saliency Map

Besides the numeric heatmap quality metric AOPC we just looked at, there is another way to verify the soundness of the heatmap, which is Sanity Check for Saliency Map proposed by Adebayo and his colleagues [14]. It offers us a binary result whether a heatmap passed the Sanity Check or not. In this paper, the experiments were mostly done on Gradient based methods. As Guided Back Propagation methods gained many attentions and success nowadays, Dr. Leon Sixt and his colleagues conducted the sanity check on most of the Guide BP methods [15] and you can find the results in Figure 3.

The intuition behind this method is that if the parameters of the original model are randomized, the produced saliency map should change. The parameters randomization is performed in cascading fashion from top to bottom layer, with the last Dense layer as top layer and the convolutional layer right after input layer as bottom layer. After we randomize the parameters in the top layer, we generate heatmap. Then randomize the second layer parameters on top of the first layer randomized model and generate corresponding heatmap. Repeat the same process until the bottom layer is reached.

We can find the above-mentioned process described in Figure 3 part (a) on the horizontal axis. On the vertical axis, we can see that 10 methods are tested. Apparently, we would expect the heatmap to change, since the top layer directly makes final decision.

Surprisingly, most saliency maps stay identical or only switched signs, which means most of them failed the sanity check. Therefore, these methods do not explain the networks' predictions faithfully. In the lower part of Figure 3, it is to test the Class insensitivity of LRP-α1β0 on VGG-16. Part (b) is the original input image where you can see a Persian cat on the left and a King Charles Spaniel dog on the right. Part (c) is the explanation for cat and part (d) is the explanation for dog. Surprisingly, you can see they are almost the same, which means the saliency maps becomes visually identical for different classes. It is concerning because the generated heatmap is not faithful to the label it is trying to explain.

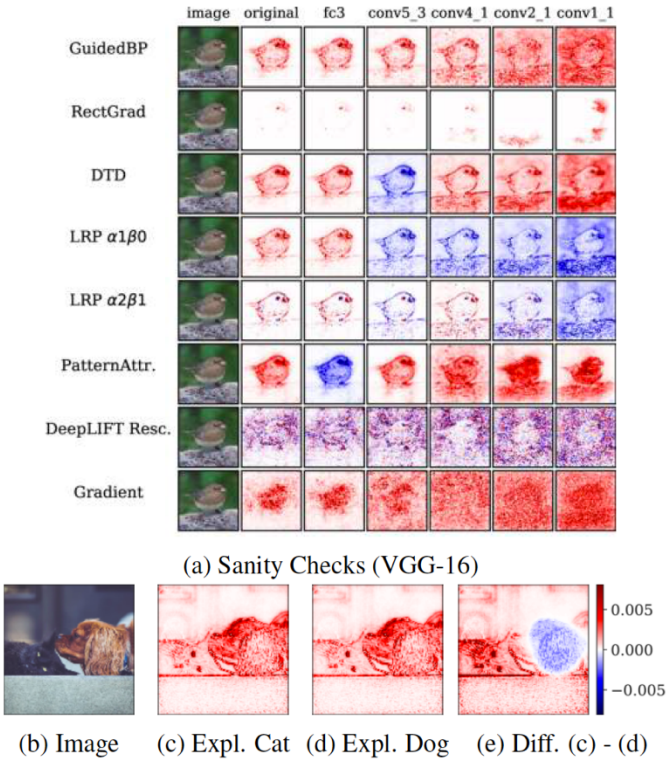

Figure 3: Sanity Check for Guided BP methods [15]

What caused these widely used heatmap methods failed the saliency test? In this publication [15], they not only raised the problem but also shed light on the cause of it. As we know in the back-propagation process, the relevance matrix is propagated layer by

layer. Therefore, by the time the relevance score matrix reaches to the first input layer, it is the result of a matrix multiplication chain. In some back-propagation rules, such as LRP-$\alpha 1\beta 0$, only the positive contributions are used. Therefore, the final relevance score matrix is the result of a positive matrix multiplication chain, and at certain layer, the result converges to a rank 1 matrix.

Here is geometric proof for the matrix convergence. On the left side of Figure 4, we can see a convex cone in light blue. Inside of it, the light red convex cone consists of all vectors $\alpha x + \beta y$ with α and β being positive, for the depicted x and y vector. It shows that all non-negative linear combinations of x and y fall into this light red convex cone. On the right side of Figure 4, we can see the similar illustration for column vectors convergence in the backpropagation. As we mentioned, the matrices are all non-negative due to the rules we choose, so the column vectors are in the positive quadrant, and all non-negative linear combination of the column vectors will remain in its convex cone. It ensures that the cone shrinks with every iteration and it converges to a single vector in the end.

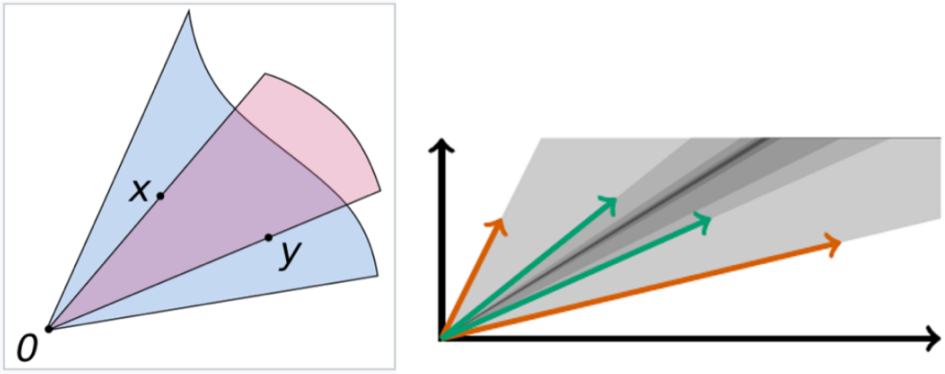

Figure 4: Convex Cone Illustration for the Convergence [15]

As we know, the rank of a matrix represents the max number of linearly independent column or row vectors in the matrix. If a matrix has rank 1, then it can be written as the product of two column vectors. For example, in Formula 5, we have a rank 1 matrix C that is rewritten as the multiplication of two column vectors c and γ. If we multiply C with any vector ν, the new vector ν can be combined with the transpose of γ and results in a constant λ that only changes the scaling. But the direction of the resulting vector stays identical.

$$C = c\gamma^T \text{ where } c \in \mathbb{R}^n \text{ and } \gamma \in \mathbb{R}^m$$
$$C\boldsymbol{v} = c\gamma^T\boldsymbol{v} = \lambda c \text{ with } \lambda \in \mathbb{R} \qquad (5)\ [15]$$

Therefore, once a matrix has converged at layer k, any layers after that do not contribute to the final result other than scaling. But the scaling is irrelevant since the heatmap is normalized in the end. As we know, the final decision of a network is made in the last layer which is after layer k. It doesn't contribute to the explanation other than scaling because the matrix has already sufficiently converged. That's why the saliency maps become visually identical for different classes. In summary, the positive contribution back propagation rule yields a multiplication chain of non-negative matrices which converge to a rank 1 matrix, that is the reason why these heatmap methods failed the sanity check.

CHAPTER 3

Experiments and Results

Chapter 3.1 Datasets and Setups

Open Access Series of Imaging Studies (OASIS) is a project aimed at making neuroimaging dataset of brain freely available to the scientific community, by providing open access to a significant database of neuroimaging and processed imaging data across a broad demographic, cognitive, and genetic spectrum. All data is available via www.oasis-brains.org. In this thesis, I used Open OASIS dataset-2 [8], which contains longitudinal MRI data in no demented and demented older adults. 150 subjects aged between 60 to 96 participated in the study and each one was scanned on at least two visits. There is at least one-year gap between visits and 373 imaging sessions were collected in total. For each single scan sessions, 3 or 4 individual T1-weighted MRI scans were obtained. All subjects are right-handed and both men and women are included. Out of the 150 subjects, 72 were characterized as nondemented throughout the study; 14 were characterized as nondemented at their initial visit and were subsequently characterized as demented at a later visit; 64 were characterized as demented at their initial visit and remained so for subsequent scans. Both 2D and 3D MRI images are provided in this dataset. We started with the 2D images but realized the CNN model suffers from low performance due to the loss of 3D spatial information. Thus, in this thesis, we focused on the 3D version where each image is in the size of 128 x 128 x 128 voxels after preprocessing. The MRI image size is not small, but given enough time and resources, there is no problem to train a model with decent accuracy.

In this thesis, we are interested in not only obtaining a model with high classification accuracy but also understanding how the model makes decisions.

To verify the observations from the OASIS-2 dataset, we also conducted all the parallel experiments in the 3D MNIST dataset from Kaggle (https://www.kaggle.com/daavoo/3d-mnist). Just like the original 3D MNIST dataset, it is a dataset of handwritten digits but in 3D format. The data is provided in the format of point cloud, which is a set of data points in space produced by 3D scanners through measuring points on the external surfaces of objects. With some preprocessing scripts, I generated voxels from the original cloud points so the CNN model can be trained on this dataset. The entire dataset contains 5000 training images and 1000 testing images, in the size of 32 x 32 x 32 voxels. Due to the smaller size comparing to OASIS-2 dataset, it takes shorter time to conduct tests on this dataset. Since this is a handwritten digit dataset whose input images we are all familiar with, it is easier for us to interpret the corresponding heatmaps and observe patterns, which can be used as comparison and confirmation for the results from OASIS-2 dataset.

In the previous section, we saw many kinds of deep learning models have been designed for the AD research. Some directly used autoencoder in multi-modalities to extract generic features, some use GAN to generate missing modality data so multi-modalities could be used, some used transfer learning so features learned on AD can be transferred to MCI, but in this thesis we discovered that for AD vs NC classification task, a sequential model with VGG-16 architecture suffices to achieve above 95% accuracy. But as we have mentioned, our focus is more on understanding 3D CNN's decision than achieving high accuracy. To

explain the model's decision using Class Activation Mapping (CAM) method, we need to remove the last two Dense layers, because as claimed in the CAM publication the fully connected layers cause the network to lose the spatial features that it has learned in the early stage. Thus, fully connected layers are removed in the model. It causes the parameters in the model to decrease and so does the model complexity. As a result, the prediction accuracy on the testing data also dropped, but it can still be considered as decent and the details can be found in the coming paragraph. Besides CAM, we also used another heatmap method Layer-wise Relevance Propagation (LRP) to visualize the network's decision. To compare these two heatmap methods equally, same model is used.

As we know, overfitting is not an uncommon issue for 3D CNN, especially in the AD classification task where relatively small amount of training samples is available but large number of the parameters need to be trained in the model. There are several techniques to address this issue, such as drop out, weight decay, early stopping, etc. In this thesis, early stopping is the technique we chose. For both datasets, 10% of the training data is used as validation set to prevent overfitting. The patience is set to be 3, which means if the validation loss set keeps decreasing over three epochs, the training process will be stopped. By applying this technique, we can prevent the model from just memorizing the data instead of learning general features. For the optimizer, Adam is used with the learning rate of 0.00001. For the loss function, categorical cross-entropy is used for both since they are classification tasks. For the OASIS-2 dataset, even though binary cross-entropy could be used, we stick to the categorical cross-entropy so the implementation for OASIS-2 can be generalized to 3D MNIST dataset.

In the training process, usually people dedicate a bigger part of data as training and a smaller part as testing, for example, 80% and 20% correspondingly. As we can imagine, after training there might be some biases in the model since it only extracted features from 80% of the data. What if some important features happen to only exist in the 20% testing data? Then the model obtained will not be able to generalize well to unseen data in the future and will also have poor performance on the testing data. Another case could be all features in the testing data happen to be well learned already but not so for all features in the training data. In this case, the model will have a very high prediction accuracy but it's misleading. Thus, to ensure the model is not biased to specific part of the training data, we used K Fold Cross Validation in the training process. The overall illustration is depicted in Figure 5. It starts with diving the data into K subsets. There will be K rounds in total and at each time, one of the K subsets is used as the test set and the other K-1 subsets are combined together to form the training set. The average performance of all iterations is considered as the overall performance. As we can see, every data point gets the chance to be trained for k-1 times and be tested for once. Therefore, it significantly reduces bias and variance as all data has been looked at for pattern extraction.

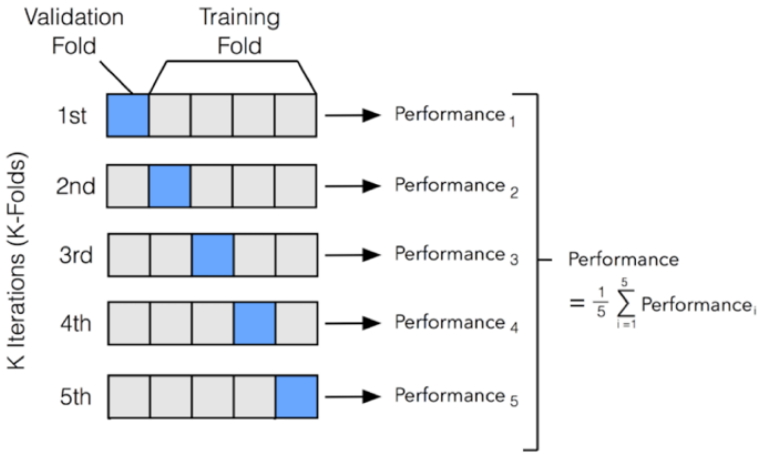

Figure 5: Cross Validation Illustration

In this thesis, a 5-fold cross validation is used for the 3D MNIST dataset and 10-fold cross validation is used for OASIS-2 dataset. The testing performances are listed in Table 3 and Table 5. As we know, the standard deviation is a measure of the variation amount in a set of values, which can be used to check if the model performance is stable or not. Thus, mean and standard deviation are calculated for both datasets. The corresponding model architectures are listed in Table 2 and Table 4.

Table 2: 3D MNIST Models Architecture

Number of Convolutional Layers	Architecture
2	Conv, Maxpool, Conv, GAP, Dense
3	(Conv, Maxpool) x 2, Conv, GAP, Dense
4	Conv, (Conv, Maxpool) x 2, Conv, GAP, Dense
5	(Conv, Conv, Maxpool) x 2, Conv, GAP, Dense
6	(Conv, Conv, Maxpool) x 2, Conv, Conv, GAP, Dense

Table 3: 5-Fold Cross Validation Testing Result for 3D MNIST Dataset

Number of Convolutional Layers	1	2	3	4	5	Mean	Std
2	56.5	58.42	56.58	53.17	54.92	55.92	1.77
3	85	87.08	84.83	85.42	86	85.67	0.81
4	90	89.5	88.58	88.67	90.42	89.43	0.72
5	90.33	91.92	91.75	90.67	92.83	91.5	0.90
6	93.92	95.92	95.83	95.42	95.58	95.33	0.73

Table 4: OASIS-2 Models Architecture

Number of Convolutional Layers	Architecture
3	(Conv, Maxpool) x 3, GAP, Dense
6	(Conv, Maxpool) x 6, GAP, Dense

Table 5: 10-Fold Cross Validation Testing Result for OASIS-2 Dataset

Number of Convolutional Layers	1	2	3	4	5	6	7	8	9	10	Mean	Std
3	52.9	52.9	52.9	50	52.9	50	52.9	50	50	47.1	51.2	1.92
6	88.2	82.4	88.2	94.1	88.2	85.3	94.1	91.2	88.2	94.1	89.4	3.76

Chapter 3.2 CAM for 3D MNIST and OASIS-2

As we introduced in the previous chapter, CAM is the heatmap method that generate the relevance scores by using the class activation map, which is calculated using the weighted sum of the feature maps from the last layer before global average pooling. Then simply up-sample the class activation map to the original image size and save it as heatmap. Considering the simplicity of this method, will it work well for complex models? How would the heatmaps change under different level of model complexity and different datasets? Can we trust the heatmaps to always faithfully present the features that the model uses to make decision? In this section, we seek to answer these questions by experimenting on 3D MNIST and OASIS-2 datasets using well-trained models with different architectures. In the training process, k-fold cross validation is used to verify the stableness of the model performance, and the related details can be found in section above.

In both datasets, heatmap is generated for every single image in the testing set. To make sound statements, hand picking heatmaps to show could be biased. Directly using the average of all heatmaps for the same class could be one way of solving the cherry-picking issue. But the uniqueness of each input leaves the simple average result hard to understand. Take the 3D MNIST dataset for example, there are many ways for human to write a digit 4. Some like to close the top part, and some like to leave it open. Thus, it is likely that the network will rely on different features in different input to make the decision. Simply taking average over all heatmaps will cause us not to be able to recognize the real features that is highlighted, since the heatmap and input are misaligned. This is the typical issue that could be addressed by the image registration technique, which is to transform different

sets of images into one coordinate system. This part is kindly contributed by my dear lab mate Ankita Prashant Joshi. Due to the time and computation resource limit, we only registered heatmaps on the 3D MNIST dataset. For each digit from 0 to 9, we registered around 10 source images into their corresponding target images using a chosen templet, then use the average of all registered images. In both CAM and LRP, the heatmaps for 3D MNIST presented in this thesis are registered. Even though we were not able to register the OASIS-2 dataset, we did observe similar patterns in both datasets, which we will illustrate in detail in the following sections.

Chapter 3.2.1 CAM for 3D MNIST

On the left side of Figure 6, we can see a summary of the models' complexity change. As the number of layers increases, the parameters in the model increase and the model gets more and more complex. We know that as the network gets deeper, the low-level features captured in the first few layers will be passed into higher layers for further abstraction. Therefore, the network can capture features at different level and the performance should also increase. On the right side of Figure 6, we do see this expected trend that the prediction performance increase monotonically when the model gets more and more complex.

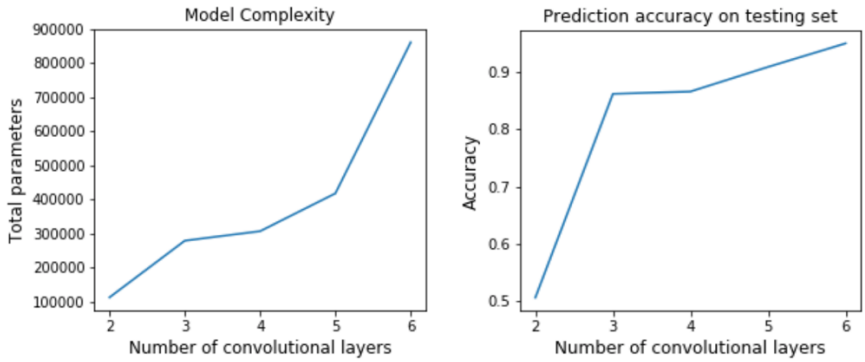

Figure 6: Model Complexity and Prediction Performance Change in 3D MNIST Models

The results for heatmaps generated by the simple model with only 2 convolutional layers is displayed in Figure 7. All three views are available for the heatmap, but here we only show the plane that best reflects the digit pattern for the sake of space. In this heatmap set, we can see that in most cases, the model is using the overall structure of the digits to make decisions, since the heighted features follow the stroke closely. For example, for digit 0 and 8, the highlighted regions are exactly the corresponding strokes. For digit 1, the absence of other strokes plays an important role for the model to make prediction.

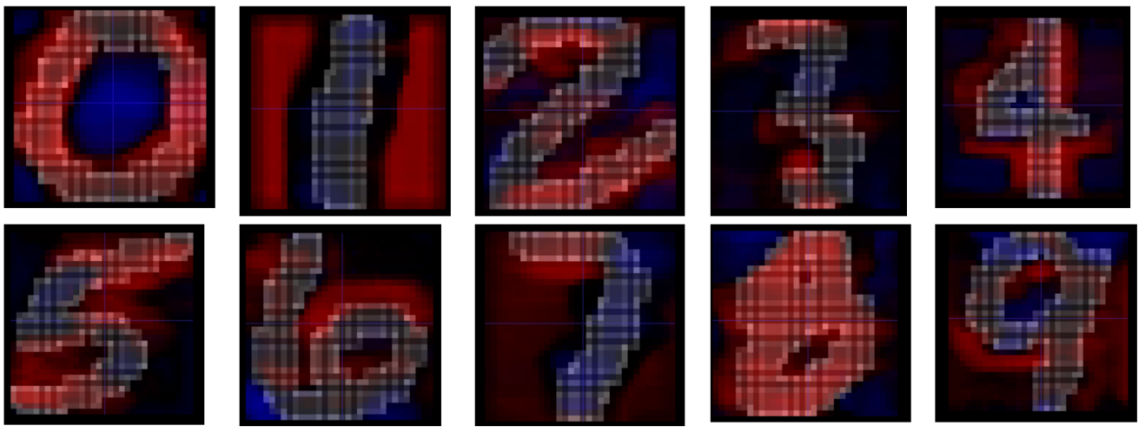

Figure 7: Registered 3D MNIST Heatmaps Using CAM in the Model with 2 Convolutional Layers

In Figure 8, we can see the heatmaps generated by the complex model which has 6 convolutional layers. Comparing to Figure 7, the model doesn't rely on the overall contour to make decisions anymore. The highlighted features are more abstract and for some digits it makes sense visually but not all not them. For digit 0, we can see the center of the circle is highlighted and it shows that closed circle is very important to predict label 0. For digit 7, the highlighted region is in upper left. It makes senses since that's the difference between 7 and 9. For digit 3, the middle part is highlighted. It also makes sense since it is how we distinguish 3 from 8. But for other digits, it is hard to make sense of them directly.

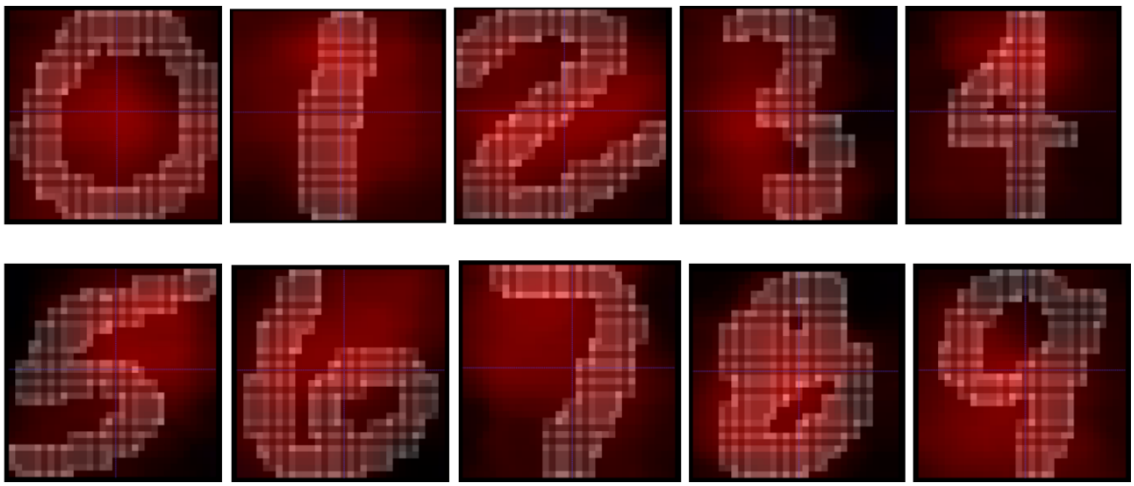

Figure 8: Registered 3D MNIST Heatmaps Using CAM in the Model with 6 Convolutional Layers

Chapter 3.2.2 CAM for OASIS-2

Same with the MNIST, we also generated heatmaps from both the simple and complex model in OASIS-2 dataset. The heatmap patterns are actually very similar across all subjects. Thus, showing result for one subject is enough and here we chose subject 28. In Figure 9, we can see the corresponding transverse, sagittal and coronal views of the heatmap generated by the simple model. The heatmaps are in RGB color scheme where the red channel and blue channel represent the positive and negative contributions correspondingly. Part of the cerebral cortex and ventricle are highlighted as positive contribution and these areas could be where lesions are detected. In the rest part of the MRI image, the overall structure of the brain is highlighted as blue. It could be for this subject that the network treats these parts as normal. Thus, it contributes negatively for the model to make the decision about AD.

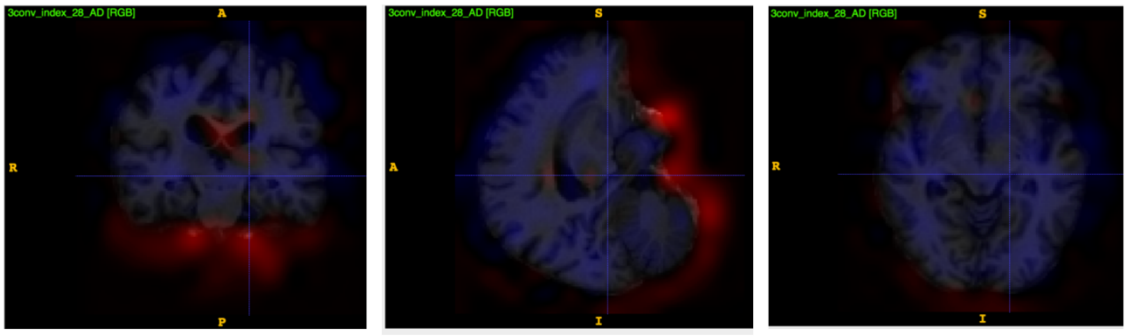

Figure 9: Heatmap for Subject 28 in OASIS-2 Using CAM in the Model with 3 Convolutional Layers

With a more complex model, the prediction performance also increases for the OASIS-2 dataset. And how about the corresponding heatmap? Similar to the 3D MNIST results, the heatmap stopped to make sense visually. As you can see in Figure 10, even some corners where there are no voxel values are highlighted as important for the model to make decision. Thus, in this brain MRI image dataset, we can't rely on CAM to visually show us the important features in the complex model.

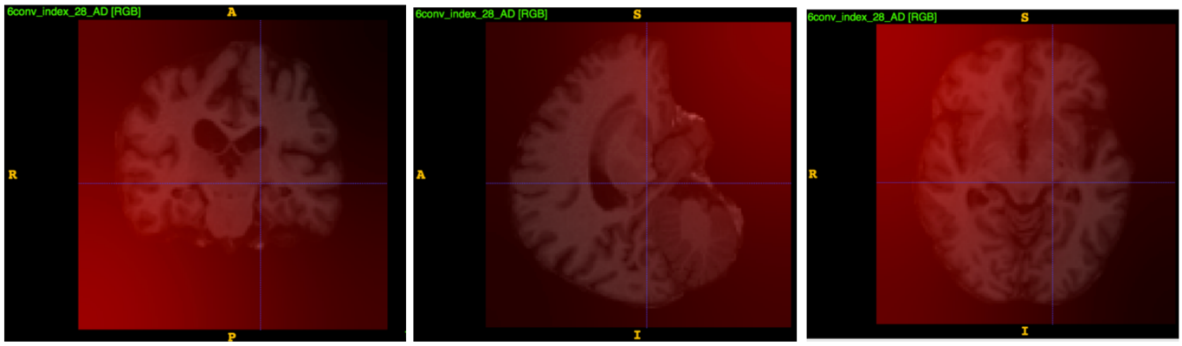

Figure 10: Heatmap for Subject 28 in OASIS-2 Using CAM in the Model with 6 Convolutional Layers

By looking at these heatmaps from both datasets, we can see that when the model is less complex, it relies more on the contour of the input images to make decision. The prediction accuracy is not high due to the lack of abstract features. But the heatmaps generated by CAM using simple model can visually explain what features the model used to make decision. On the other side, when the model gets more complex, the prediction accuracy

will increase as expected. But The highlighted regions make less sense visually. This pattern is verified in two datasets, and we see it as the limit of CAM method. Due to the simplicity of its design, it couldn't convert and transform the abstract features to visual representation accurately when the model gets complex.

Chapter 3.3 LRP for 3D MNIST and OASIS-2

As we mentioned in the illustration of LRP in Chapter 2.3, we know that this method works by back propagating the final prediction score layer by layer using the contribution percentage during the forward propagation. Finally, when the relevance score reached the input layer, it will be saved as the heatmap for corresponding prediction. There are many LRP rule designed for different purposes. It is suggested a composite rule should be used so that we can make the most of each rule's advantage and avoid its disadvantage. From Table 6 we know that LRP- Z^{β} should be used for the last layer which is the layer right before the input layer. LRP-γ or LRP-$\alpha\beta$ should be used for the lower layers. LRP-ε should be used for middle layers and LRP-0 should be used for the upper layers. However, there is no clear cut between the definition of lower, middle, and upper layers. It is for sure that the 2^{nd} layer, the 4^{th} and the 6^{th} layer should be considered as lower, middle and upper layer correspondingly. But the 3^{rd} layer could be treated as either lower or middle layer and the 5^{th} layer could be thought as either middle or upper layer. For the Max Pooling layer, we could use either winner-take-all strategy or average pooling strategy. The values for γ, ε, α, β should also be tuned for different model and dataset. Thus, considering all of the combinations, the composite rule space we can explore is actually infinite. Due to the time

limit, we can only explore a small set of composite rules. In this thesis, the three rules we chose are listed in Table 6.

Table 6: Composite LRP Used in This Thesis ($\gamma = 0.25$, $\varepsilon = 0.0025$, $\alpha = 1$, $\beta = 0$)

	1st Conv	2nd Conv	3rd Conv	4th Conv	5th Conv	6th Conv	Maxpooling
Composite LRP 1	LRP-z^β	LRP-γ	LRP-ε	LRP-ε	LRP-0	LRP-0	Winner-take-all
Composite LRP 2	LRP-z^β	LRP-γ	LRP-γ	LRP-ε	LRP-0	LRP-0	Average Pooling
Composite LRP 3	LRP-z^β	LRP-$\alpha\beta$	LRP-$\alpha\beta$	LRP-ε	LRP-0	LRP-0	Average Pooling

In the previous CAM section, different models are used to explore how the heatmap method behave in different model complexity. In LRP, we are more interested in understanding how different LRP rules affect the heatmap. At the same time, we know that LRP is a sophisticatedly designed heatmap method that carefully pass the relevance score using weights as guidance. It is less likely to have a big difference when dealing with models at different complexity. Thus, we only experimented on the complex model. To be able to make fair comparisons between these two heatmap methods, we generated heatmaps from the same model in LRP and CAM.

Chapter 3.3.1 LRP for 3D MNIST

In the 3D MNIST heatmaps generated by composite LRP 1 and 2, only small regions are highlighted. Thus, they are not very helpful in explaining the network decision. For the sake of space, we only show the heatmaps from composite LRP 3 in Figure 11 which have more significant results. We can see that the features highlighted are very specific to individual input. For digit 0, the middle part on the right side of the contour is highlighted.

It makes sense because with an additional stroke there, it will become 9. For most of other digits, the features are highlighted in a very similar fashion. For example, the bottom left part highlighted in digit 7 distinguishes itself from digit 2. The upper left part highlighted in digit 4 makes the difference with digit 9. Also, the middle part highlighted in digit 3 is important for it to be a 3 instead of 8. The bottom right part highlighted in digit 1 makes it different with digit 7. The way digit 5 is written makes it become 6 if an additional stroke is added near the highlighted part. For digit 8, the way it is written makes the first circle very important for it to be a digit 8, since that circle almost becomes solid instead of hollow. It makes sense to highlight that part.

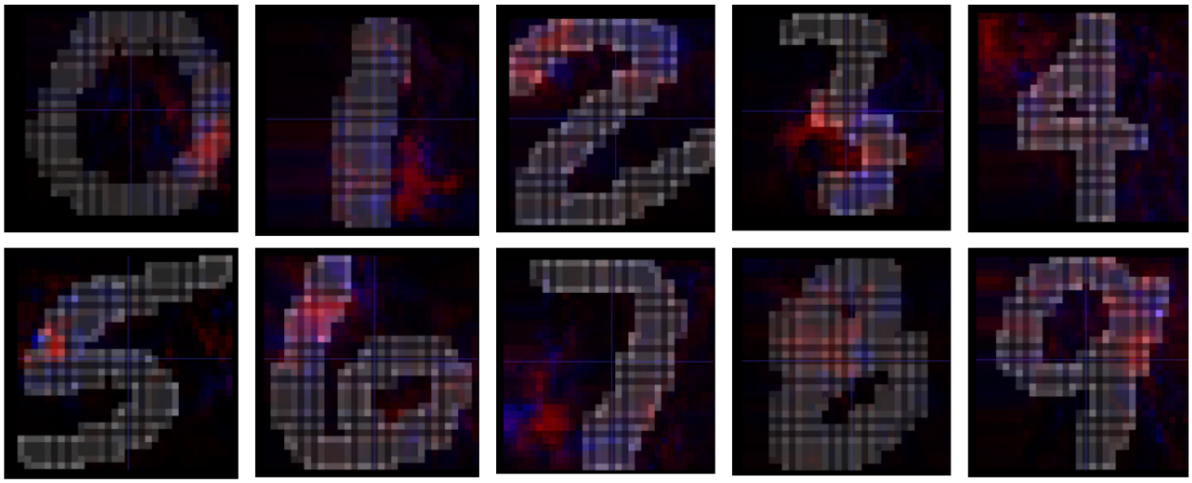

Figure 11: Registered 3D MNIST Heatmaps Using Composite LRP 3

Chapter 3.3.2 LRP for OASIS-2

In this section, we use LRP methods to explain the 3D CNN model's decision about Alzheimer's Disease. This is not the first effort in this field. Dr. Bohel and his colleagues have already published their results in Frontiers in Aging Neuroscience at 2019 July [12]. However, the efforts in this thesis are independent from their work. It means the overall

pipeline design, code implementation and LRP rules setting are of our original work, since we were not aware of this publication until the middle phase of our exploration. But we did get inspirations from their work about how the LRP-αβ rule is understood in AD classification context, which we will see in the coming section.

Same with 3D MNIST dataset, the heatmaps generated by composite LRP 3 shows the strongest signal while the heatmaps from the other two rules are relatively weak. Therefore, we only show the heatmaps generated by Composite LRP 3 for subject 16, 24 and 28 in Figure 12, 13 and 14. For AD subjects, we can observe the highlighted areas are very specific to individual subject in terms of location and intensity. At the same time, we can also see there are similarities in the highlighted regions across different subjects, which is that most of them are in the cerebral cortex in the frontal lobe and temporal lobe. It matches with what is known about the brain legions related to Alzheimer's Disease. For Normal Control (NC) subject, the overall structure of the brain is used to make decisions and the heatmap intensities are also relatively low. Thus, we can see that heatmap generated by composite LRP 3 have great potential in identifying the features the model used to classify an input as AD. More importantly, it is able to give specific explanation for a subject.

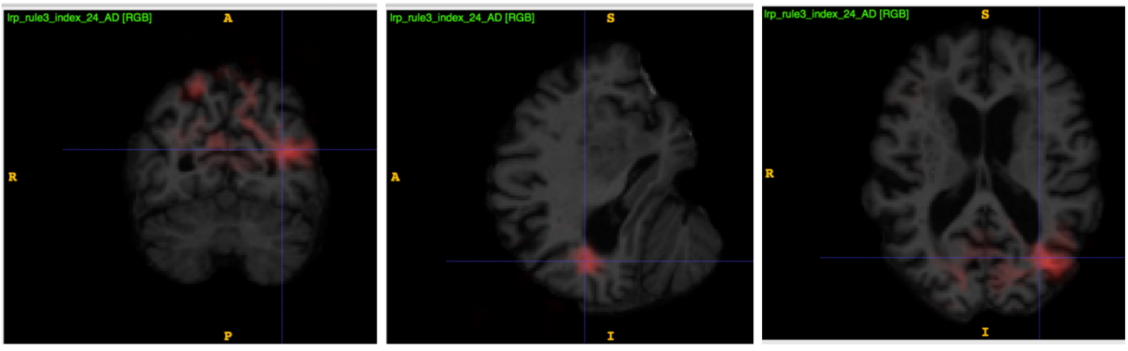

Figure 12: Heatmap for AD Subject 24 in OASIS-2 Using Composite LRP 3

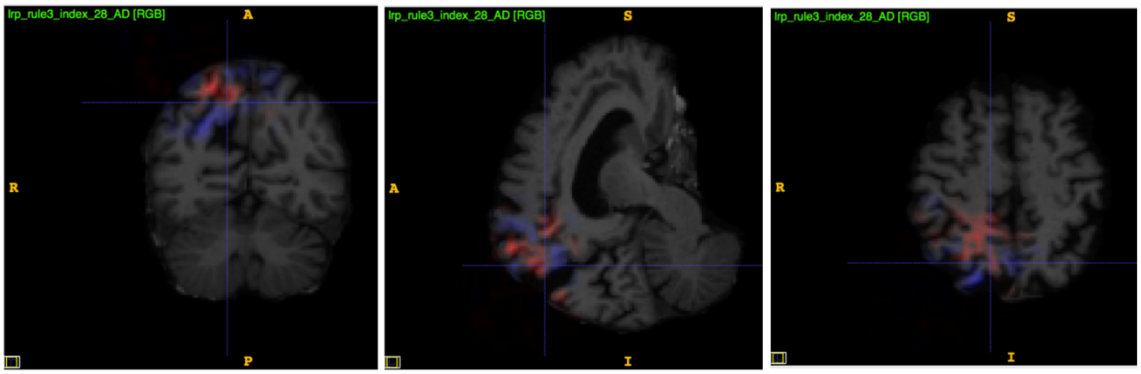

Figure 13: Heatmap for AD Subject 28 in OASIS-2 Using Composite LRP 3

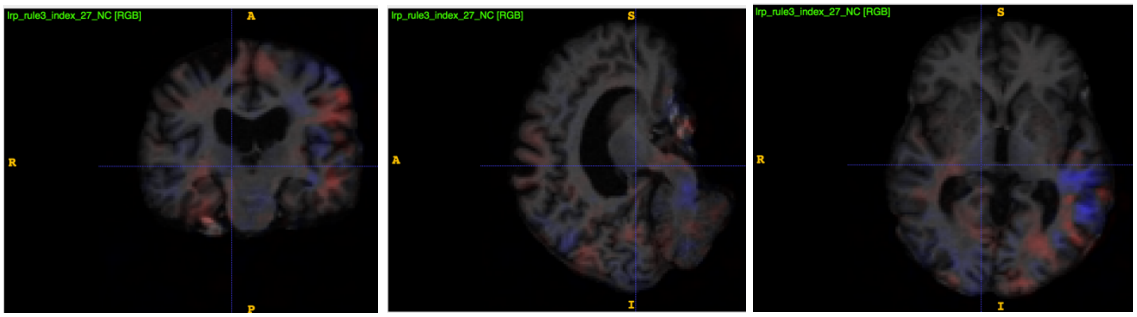

Figure 14: Heatmap for NC Subject 16 in OASIS-2 Using Composite LRP 3

In the heatmaps results from 3D MNIST and OASIS-2 dataset, we can see that composite LRP 3 achieved the best performance in identifying useful features for the model. The difference between composite LRP 3 and the other two rules is that LRP-αβ is used for the lower layers and Average Pooling is used for the Max Pooling layers during the back propagation. From the LRP rules summary in Table 6, we can see that both LRP-αβ and LRP-γ promote the positive contribution. But LRP-αβ also inhibits the negative contribution. In composite LRP 3, β is set to be 0, which means the negative contributions are totally eliminated, and only the positive contribution during the forward propagation plays a role in distributing the relevance scores. In the LRP in AD classification paper [12], they did experiment on different β values and found the sparseness increases with higher β value. When β is close to 0, the network focuses on the positive contribution and is more

clinical interpretable. In the LRP Overview paper [11], they used average pooling strategy in back propagation for the Max Pooling layer, but no justification is given about their choice. In our experiments, it turns out the average pooling strategy also works better than the other one, but we haven't figured out the reason yet. Therefore, at this point, we discovered that composite LRP 3 have great potential in explaining the network's decision at an individual level. But more work is still needed to reveal why the average pooling is superior than the winner-takes-all strategy for the Max Pooling layer.

Chapter 3.4 AOPC for 3D MNIST and OASIS-2

So far, we looked at the heatmaps generated by CAM and LRP on two datasets. By visual appearance, we can make judgement call about which heatmap is better based on our opinions. But different person might have different opinions. Sometimes, the difference between two versions of heatmap are not easily distinguishable by naked eyes. Thus, an objective metric that can quantify the heatmap quality is needed.

Area over Perturbation Curve is proposed for such purpose. As we mentioned in the previous introduction chapter, the first step to use this metric is to divide the input image into unisize cubes. There are two strategies we can choose. First one is to use a predefined grid to divide the image so that the cubes are adjacent to each other, which we call uniform division strategy. The second one is to use greedy search fashion to always pick the cube that returns the greatest heatmapping function value, which we call greedy search strategy. The heatmapping function h_p also needs to be designed. One very intuitive function would be to use the sum of all heatmap voxel values inside a cube. In the previous heatmaps

shown, we see it is possible to have red and blue regions in the same cube, where the positive and negative signals might cancel out each other if we use a direct sum for the h_p. It will set the cube to the end of the MoRF sequence and become less likely to be chosen for perturbation. Thus, another option is to use the sum of the absolute voxel values as the h_p. In this thesis, to seek to understand how the dividing strategy, heatmapping function h_p, and stride size affect the AOPC score, we experimented several combinations of these factors in both 3D MNIST and OASIS-2 dataset.

Chapter 3.4.1 AOPC for 3D MNIST

In the 3D MNIST dataset where the image dimension is 32 x 32 x 32, the coverage percentage is listed in Table 7 under different stride size. The reason why coverage percentage matters is that the curve trend might change as more perturbations are applied, which we did observe in some of our plots. With a low coverage percentage, we could be fooled by the unstable curve in its early stage, thus, we will risk unreliable conclusion. Comparing to the 15.7% perturbation coverage that is used in the original AOPC paper, we can be much more confident about our conclusion looking at the perturbation coverage percentage in Table 7. For 3D MNIST dataset, we actually explored 5 models with different complexity, which has 2 to 6 convolutional layers in its architecture correspondingly. Even though in the previous CAM heatmap section, we didn't display heatmaps from all models for the sake of space, they are actually generated. In this section, we calculated the AOPC scores for all of them.

Table 7: Perturbation Coverage Percentage for MNIST

Stride size	Number of cubes in total	Total perturbation steps length L	Perturbation Coverage Percentage
4	(32/4)^3 = 512	500	500/512 = 96%
8	(32/8)^3 = 64	50	50/64 = 78%

In Figure 15 and 16, we can see the AOPC plots for 4 variations using different combinations of the heatmapping function h_p and the dividing strategy when the stride size is 4 and 8 correspondingly. From the top rows in these two figures, we can see that in both dividing strategies, whichever h_p function we use barely makes a difference except a very small change in the simplest model with 2 convolutional layers. If we look at the bottom rows in Figure 15 and 16, where the dividing strategy is experimented, we can see that in the first half of the perturbation steps, the scores are almost the same, but afterwards using the original voxel values gradually achieved slightly better scores. These changes don't appear to be significant actually. Comparing between stride size 4 and stride size 8, we can see using stride size 4 achieved around 10% higher AOPC scores than using stride size 8 regardless of h_p function and searching strategy setting. As we can imagine, using a smaller stride, the division is in a higher granularity and thus can associate cube priority with the features better.

Among all of the AOPC plots for CAM in Figure 15 and 16, there is one pattern that is preserved throughout every stride size, searching strategy and h_p combination. It is that the AOPC scores always correlate positively with the model complexity. Starting from simple model that has 2 convolutional layers to the complex one with 6 convolutional layers, we can see the AOPC curve for one model is always above the other if it is more complex. It

shows that the heatmaps generated by complex model have higher quality than the ones generated by simple model in this metric. In Figure 6, we saw that the prediction performance also correlates positively with the model complexity. Thus, the heatmaps generated by CAM actually correlates positively with the prediction performance. Previously, we saw CAM heatmaps that are generated from the simplest model reflect the input image contour while the ones that are from the complex model make less sense visually. But from this pattern we just mentioned, it is not the 3D Neural Network that doesn't make sense. It is the limit of the CAM method that it couldn't map back the abstract features to the original input space. This is probably due to its simplicity. If we could design a more sophisticated way for the class activation map to be up-sampled to the input image size, such as the layer-by-layer fashion in LRP, CAM could achieve better visualization results for complex models.

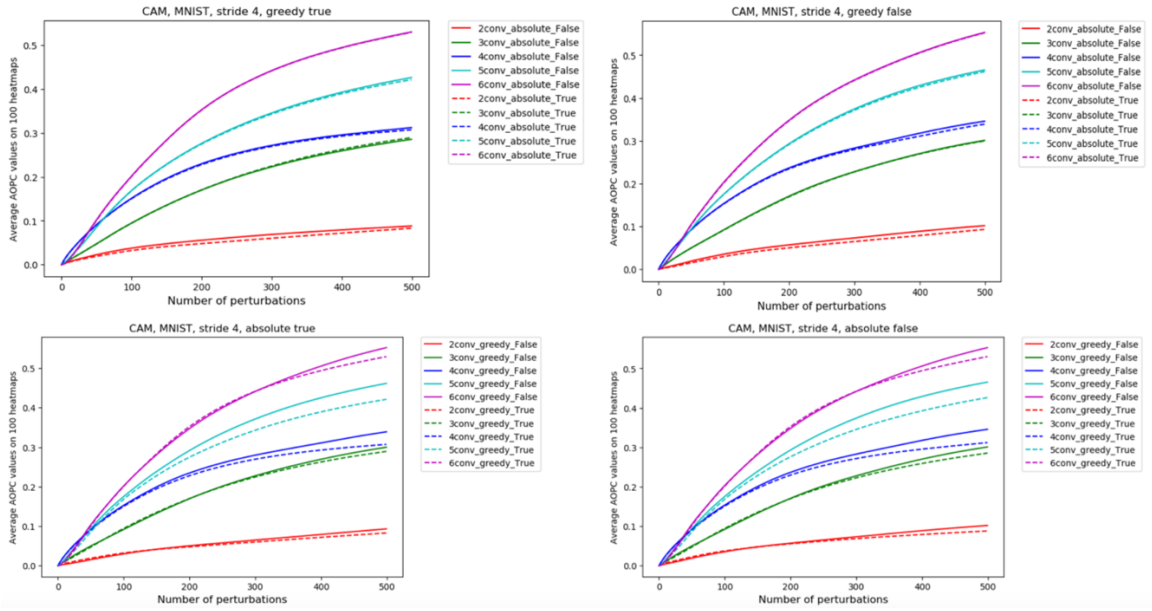

Figure 15: AOPC Curves for 3D MNIST Using CAM with Stride 4 (view in color)

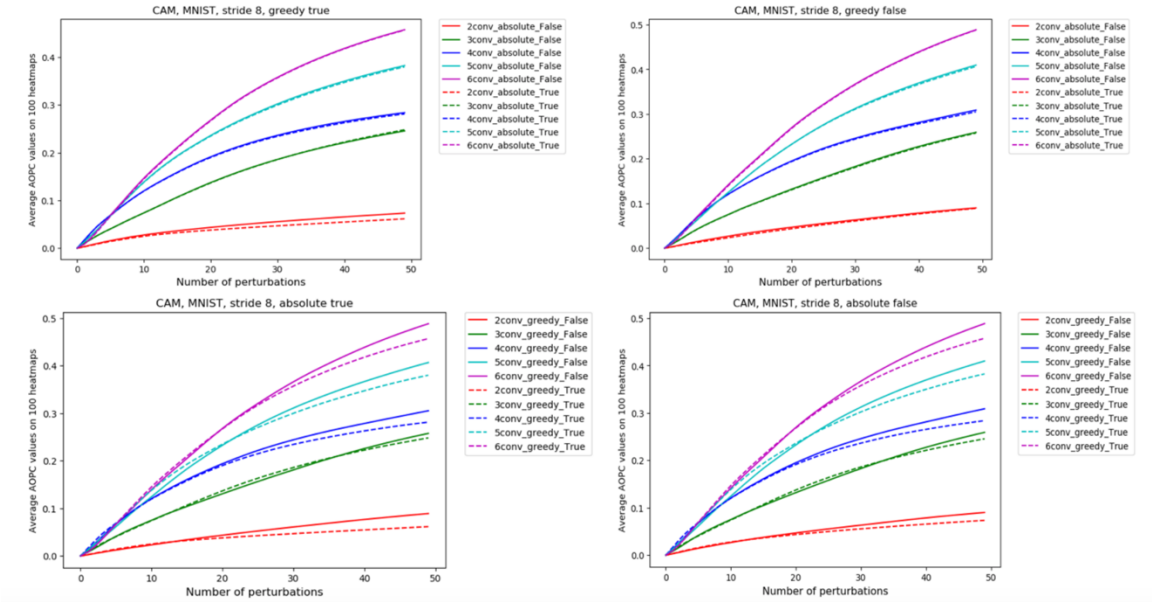

Figure 16: AOPC Curves for 3D MNIST Using CAM with Stride 8 (view in color)

The AOPC scores are generated for three LRP rules. From Figure 17, we can see all four variations when the stride size is set to 4, and in Figure 18, it is when stride size is 8. In top rows from these two figures, we investigate how the h_p function affects the AOPC scores. For composite LRP 3, we can see the difference is actually minor, which should be caused by the LRP rule that only allows positive contribution in the propagation. For composite LRP 1 and 2, as we mentioned earlier, the signals in the heatmaps are not very significant. Thus, not very much cubes in the MoRF sequence have non-zero voxel values inside. Changing h_p function will have more impact on the cube ordering in the MoRF sequence comparing to composite LRP 3, that's why we can see a big difference. From the bottom rows on Figure 17 and 18, we can see how the search strategy affects the AOPC scores. For composite LRP 3, it is almost the same though out all variants. But for composite LRP 1 and 2, using uniform dividing is better than greedy search. If we compare the final AOPC scores between stride 4 and stride 8, we can see that the scores using stride size 4 is around 5% greater than stride size 8.

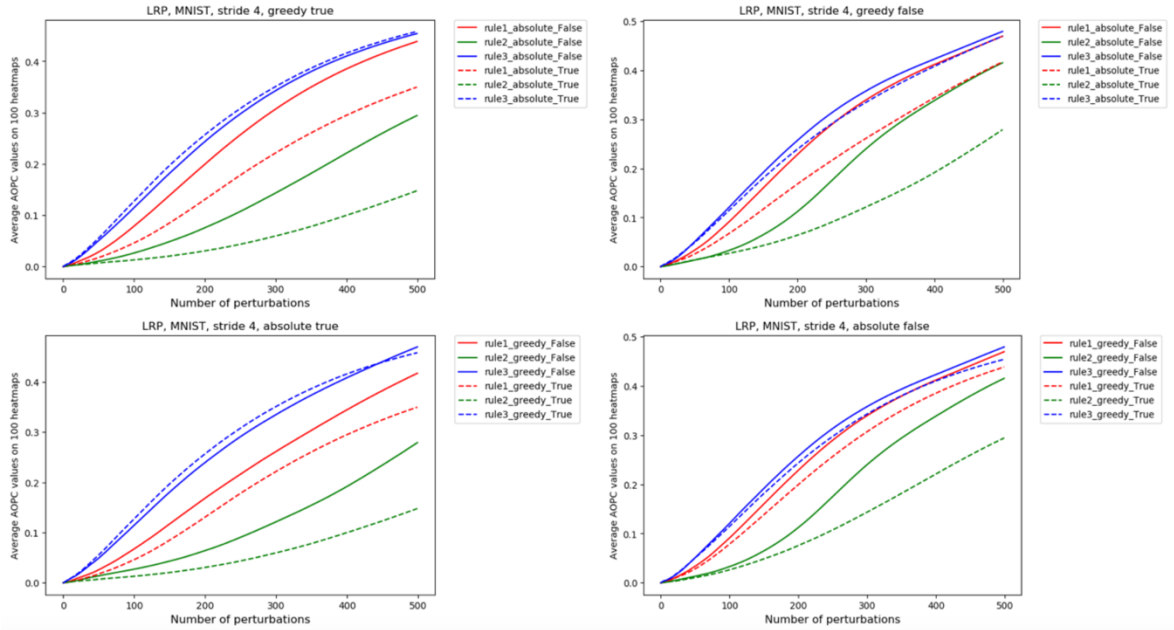

Figure 17: AOPC Curves for 3D MNIST Using LRP with Stride 4 (view in color)

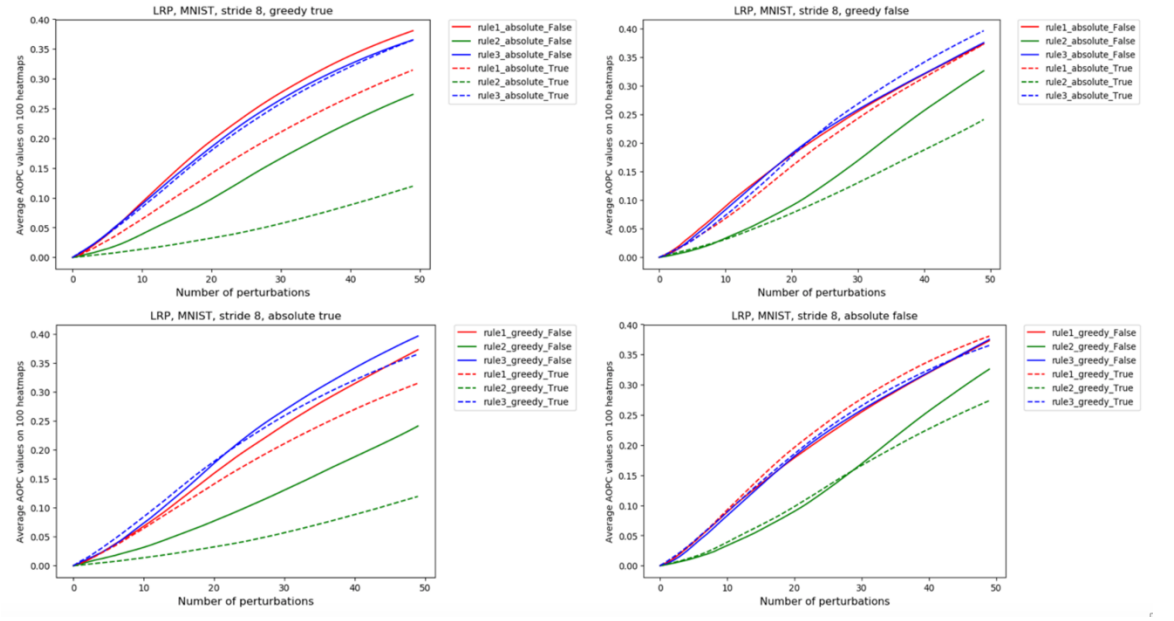

Figure 18: AOPC Curves for 3D MNIST Using LRP with Stride 8 (view in color)

From these AOPC plots for heatmaps generated by LRP, we can see that composite LRP 3 almost always achieved better scores throughout all experiments. Therefore, we conclude the heatmap generated by composite LRP 3 have the highest quality, which verifies our

previous observation that the heatmaps generated by composite LRP 1 and 2 are not significant enough to represent the features the model used to make decision. Comparing the best AOPC scores from CAM 6conv and composite LRP 3, we can see the CAM 6conv achieved around 10% greater score than LRP, which means the heatmaps are in higher quality. Even though the CAM 6conv heatmaps don't make sense visually as much as the composite LRP 3, it shows us the potential of the CAM method if we can design a more well guided process for the feature map to be up-sampled to the original input image size.

Chapter 3.4.2 AOPC for OASIS-2

Comparing to the 3D MNIST dataset that has dimension of 32 x 32 x 32, OASIS-2 dataset is in a much higher dimension 128 x 128 x 128. Therefore, it is not feasible to repeat all experiments we did in MNIST. At the same time, we see that the greedy search and a smaller stride size didn't boost the AOPC scores very much in the 3D MNIST dataset. Therefore, we only explored the h_p function factor while setting the stride to be 32 and using uniformly dividing strategy. With the total perturbation steps L set to be 50, the coverage percentage is 50/64 = 78%. Like we mentioned in the previous section, it is big enough for us to be confident about the overall curve trend.

In Figure 19 we can see the AOPC plot for CAM method in OASIS-2 dataset. If we directly use the voxel values for h_p, which means absolute parameter is set to false, the complex model initially increases slower than the simple one. But in the end, the trend is reversed, and the complex model reached higher score. As we saw in Figure 9, the cerebral cortex and ventricle is highlighted in red. Therefore, when the absolute is set to false, the cerebral

cortex and ventricle areas will get selected first. As we know the atrophy of cerebral cortex and enlarged ventricle are closely related to AD diagnosis. By perturbing these regions, the prediction score will decrease very quickly. Thus, the AOPC score will increase quickly since it measures the difference between original prediction score and the one after perturbation. That's why initially, the AOPC score for heatmaps generated by simple model grows faster than the complex one. In Figure 9, we also saw that most of the brain anatomical structure are highlighted in blue. Thus, the cubes that contains brain structure will be found at the end of MoRF sequence. Comparing to these cubes, even the ones in the corner where there are no voxel values will be chosen first. That's why after around half of the perturbation steps, the AOPC scores for heatmaps generated by complex model become greater than the simple one.

When we set the absolute parameter to be true, which means the color of the heatmap doesn't matter anymore, only the intensity make a difference in the MoRF sequence. As we just mentioned in the previous section, the heatmap generated by the simple model reflects the anatomical structure of the brain and also highlighted AD related brain legions. With absolute value h_p function, the cubes with aforementioned important features will be chosen first. That's why the heatmaps generated from simple model achieved greater AOPC scores. As we saw in Figure 10, the heatmaps generated from the complex model barely have negative highlighted regions, that’s why switching absolute parameter value doesn’t affect its AOPC scores.

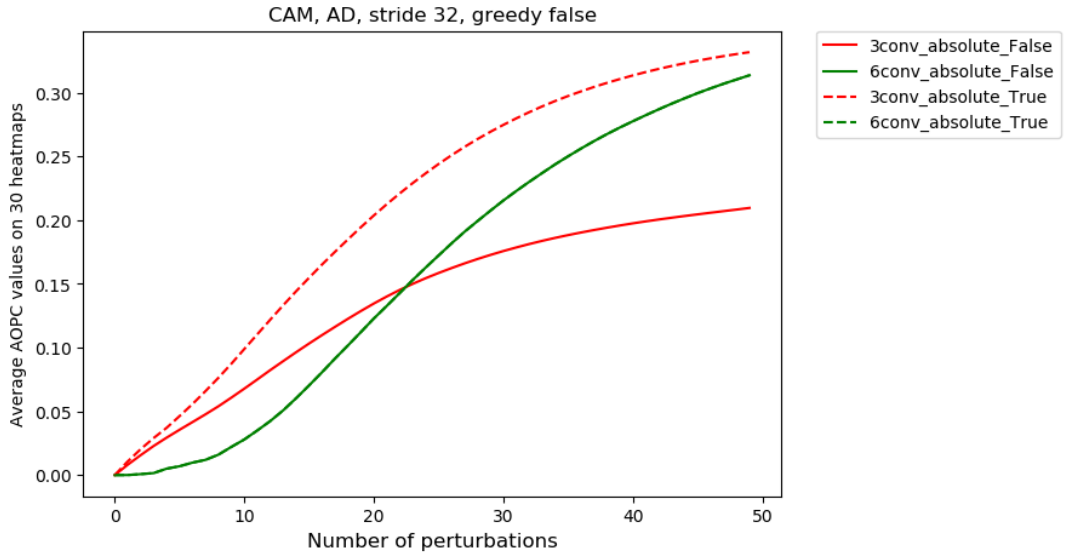

Figure 19: AOPC Curves for OASIS-2 Using CAM (view in color)

Same with CAM, stride size 32 and uniformly diving strategy are used to generate AOPC plots for the heatmaps from LRP method. In Figure 20, we can see the h_p function doesn't makes very much difference across all three rules. It is also obvious that the heatmaps generated by composite LRP 3 achieved the best score regardless of the h_p function setting, which confirms what we observed and explained in the LRP on OASIS-2 chapter.

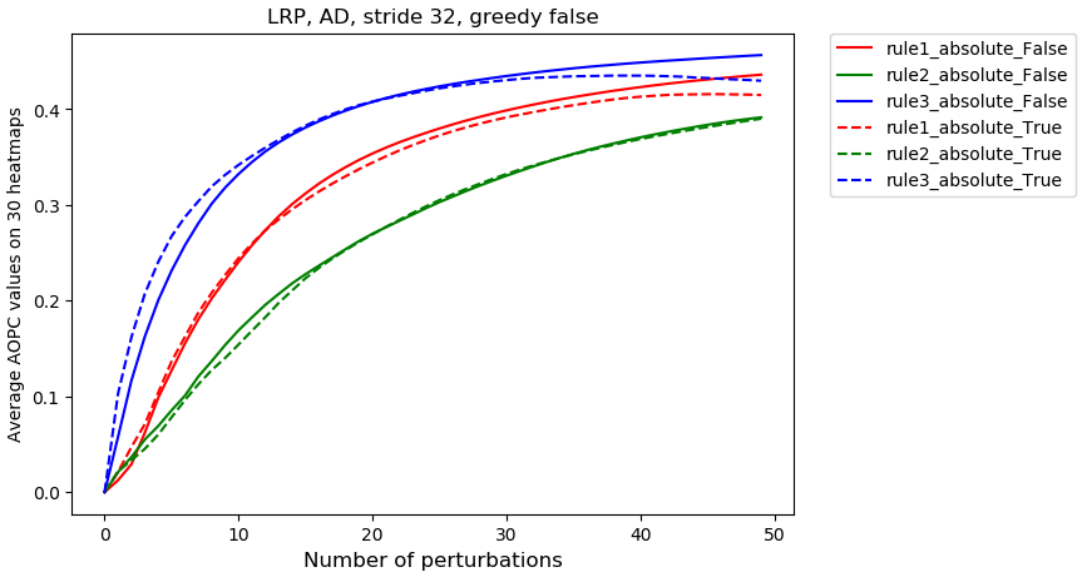

Figure 20: AOPC Curves for OASIS-2 Using LRP (view in color)

To compare the performance of CAM and LRP in explaining Alzheimer's Disease, we use the AOPC scores for the heatmaps generated from complex model, which is around 0.3 for CAM and 0.45 for LRP. We can see that the LRP method achieved around 50% higher score than CAM. Therefore, LRP have better performance than CAM at explaining the network's decision on AD.

Chapter 3.5 Sanity Check for 3D MNIST and OASIS-2

Similar to AOPC scores that evaluate heatmaps using objective metrics, sanity check is another form to evaluate the heatmap. Instead of giving a numeric measurement about the heatmap quality, sanity check offers us a binary decision whether the heatmap pass the check or not. As we introduced earlier in Chapter 2, sanity check is based on a very intuitive assumption that randomizing parameters in the model should cause the heatmap to change. To verify whether the heatmaps we generated are sane, we performed this check on both methods and both datasets. In CAM, only the second last layer which is the one between Global Average Pooling and final Dense layer has weights that are used in generating heatmaps. Thus, there is only one-layer perturbed heatmap. For LRP, since weights are needed when the relevance scores are passed through each convolutional and dense layer, we performed the parameter randomization on each layer, in the cascading fashion from top to bottom. In the models we used, there are 7 weight layers. Thus, there will be 7 versions of perturbed heatmaps in LRP.

The heatmaps generated are in 3D format and all transverse, sagittal and coronal views are available. Because the view is merely a different perspective of the same object, the

conclusion of whether passing the sanity check or not is the same across all three views. There is no need to show all of them for the sake of space. Thus, only the sagittal view for all perturbed heatmaps are displayed.

Chapter 3.5.1 Sanity Check for 3D MNIST

In Figure 21, we can see the one on the left is the original heatmap and the one on the right is the randomized heatmap. Apparently, they are different. Thus, CAM method passed the sanity check on 3D MNIST dataset.

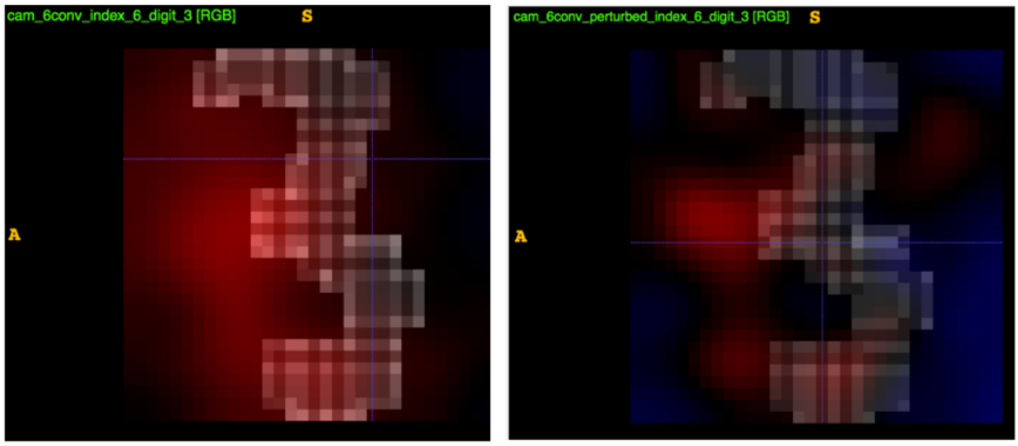

Figure 21: Original and Parameters Randomized CAM Heatmaps for Digit 3

In Figure 22, we can see the first image on the left upper corner is the original Digit 3 heatmap generated by composite LRP 3. The rest are layer parameters randomized heatmap. The randomization process starts from the top layer and increase one layer at a step till all seven layers' parameters are perturbed. We can see that each heatmap is different from its previous version. Therefore, LRP also survived the sanity check on 3D MNIST data.

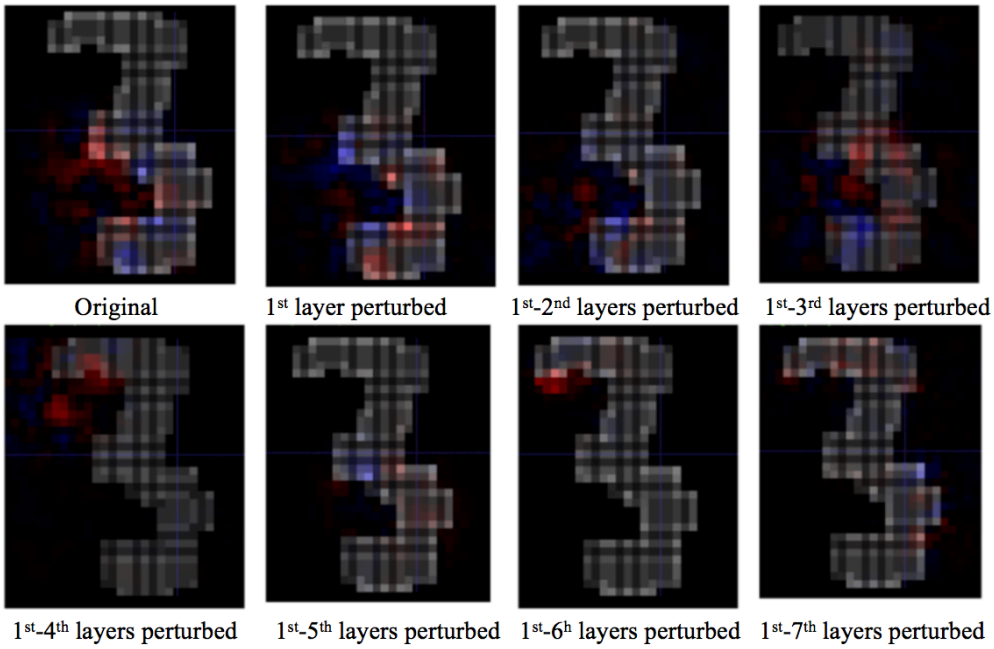

Figure 22: Original and Parameters Randomized LRP Heatmaps for Digit 3

Chapter 3.5.2 Sanity Check for OASIS-2

In Figure 23 we can see the original heatmap on the left side and the perturbed heatmap on the right side. Since this layer contribute directly to the decision making, the heatmap should definitely change. As expected, the heatmap did change. Since parameters in this layer are the only weights that CAM uses to generate heatmaps, by setting it to a random distribution, all regions in the heatmap got highlighted with almost the same intensity. Therefore, CAM survived the sanity check on OASIS-2 dataset.

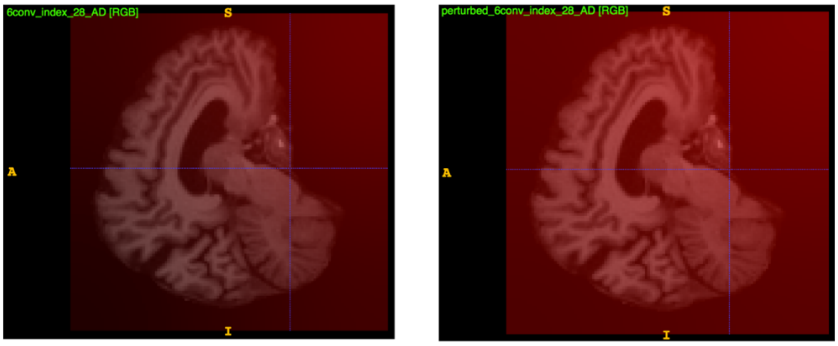

Figure 23: Original and Parameters Randomized CAM Heatmaps for Subject 28

In Figure 24, we see the original composite LRP 3 heatmap on subject 28 on the upper left corner, and then the 7 layers perturbed heatmaps are displayed on the left to right and top to bottom fashion. Immediately, we can notice that the one layer and two layers perturbed heatmap are identical with the original heatmap. Starting from the third layer, the heatmap began to respond to the parameters perturbation and begin to change. Therefore, composite LRP 3 failed the sanity check. However, that doesn't mean the heatmaps generated by composite LRP 3 is worthless. In chapter 2, we looked at the cause of this failure which is that a chain of positive matrix multiplication makes the matrix converge, and the relevance scores won't change after convergence. It is concerning that the heatmap doesn't even respond to the last layer which it uses to make decision, which could make us doubt the faithfulness of the corresponding heatmap. But in the same publication, they also mentioned that this might not be too problematic since in some cases explaining local convolutional features could be sufficient to explain a predicted class, and Alzheimer's Disease is one such case.

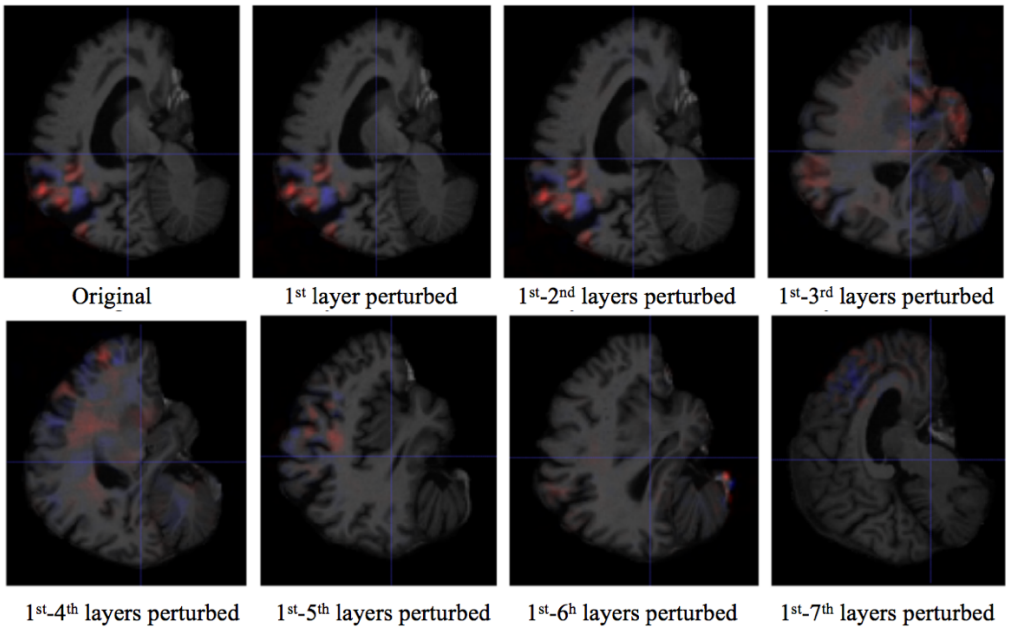

Figure 24: Original and Parameters Randomized LRP Heatmaps for Subject 28

CHAPTER 4

Conclusion

To improve the 3D CNN's transparency and explain its decision on AD classification, we started the efforts of visualizing network's prediction. Two heatmap methods Class Activation Mapping (CAM) and Layer-wise Relevance Propagation (LRP) are explored on two datasets 3D MNIST and OASIS-2. To evaluate different heatmap methods objectively, we calculated numeric scores to quantify the heatmap's quality using the method of Area over Perturbation Curve (AOPC). To further verify the soundness of generated heatmaps, Sanity Check is performed on both methods in both datasets.

In CAM, we tested how the heatmaps change in five different models with increasing complexity. We observed that in both datasets, the heatmaps generated by the simple model focuses more on the contours of the overall input image and they have higher visual interpretability. Whereas the heatmaps from complex model use the abstract high-level features to explain model's decision and it is more difficult to make sense visually. However, according to the AOPC scores, the heatmaps generated by the simple model have low qualities, and the heatmaps generated by the complex model have high qualities. Thus, the visual interpretability correlates negatively with the heatmap quality which is totally unexpected. The inconsistency is very likely to be caused by its simple design so that it can't transform the abstract feature back into the visual space. But CAM did pass the Sanity Check on both datasets, and in the 3D MNIST dataset, it achieved even around 10% greater

AOPC scores than LRP, which shows us that with a more sophisticated design to guide the feature map back, CAM has the potential to explain the network's decision well.

Comparing to CAM, the heatmaps generated by LRP in both datasets have higher visual interpretability, and they are also specific to individuals. Thus, LRP has a higher potential to bring more understanding and trust to the model's decision in AD classification task. But for some LRP rules where only positive contributions are used during the back propagation, the relevance matrix could converge at a certain layer and stop to change afterwards. Thus, in spite of the high visual interpretability and heatmap quality score in OASIS-2, this method failed the Sanity Check. Even though this doesn't mean the heatmap becomes meaningless as we explained earlier, it is concerning knowing the heatmap doesn't even change when parameters in the decision layer get randomized. However, LRP did pass the Sanity Check on 3D MNIST dataset, probably because the smaller image size slowed down the convergence rate.

When we evaluate the LRP heatmaps from the perspectives of visual interpretability and AOPC heatmap quality score, 3D CNN seems to be trustworthy in the AD classification task. However, this statement is weakened due to the fact that it failed the Sanity Check. Looking at the pros and cons of both methods, we believe new heatmap method needs to be designed so high-level abstract features can be perfectly mapped back to the visual space without suffering any aforementioned issues. In the future, as more and more efforts are devoted in this field, we believe 3D CNN can gradually become transparent and play a routine role in clinical care.

References

1. Rusinek, Henry, et al. "Regional brain atrophy rate predicts future cognitive decline: 6-year longitudinal MR imaging study of normal aging." Radiology 229.3 (2003): 691-696.
2. Morris, John C., et al. "Mild cognitive impairment represents early-stage Alzheimer disease." Archives of neurology 58.3 (2001): 397-405.
3. Schuff, N., et al. "MRI of hippocampal volume loss in early Alzheimer's disease in relation to ApoE genotype and biomarkers." Brain 132.4 (2009): 1067-1077.
4. Zhang, Yudong, et al. "Detection of subjects and brain regions related to Alzheimer's disease using 3D MRI scans based on eigenbrain and machine learning." Frontiers in computational neuroscience 9 (2015): 66.
5. Liu, Manhua, et al. "Multi-modality cascaded convolutional neural networks for Alzheimer's disease diagnosis." Neuroinformatics 16.3-4 (2018): 295-308.
6. Cheng, Bo, et al. "Sparse multimodal manifold-regularized transfer learning for MCI conversion prediction." International Workshop on Machine Learning in Medical Imaging. Springer, Cham, 2013.
7. Pan, Yongsheng, et al. "Synthesizing missing PET from MRI with cycle-consistent generative adversarial networks for Alzheimer's disease diagnosis." International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, Cham, 2018.

8. Marcus, Daniel S., et al. "Open access series of imaging studies: longitudinal MRI data in nondemented and demented older adults." Journal of cognitive neuroscience 22.12 (2010): 2677-2684.
9. Zhou, Bolei, et al. "Learning deep features for discriminative localization." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.
10. B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Object detectors emerge in deep scene cnns. International Conference on Learning Representations, 2015.
11. Montavon, Grégoire, et al. "Layer-wise relevance propagation: an overview." Explainable AI: Interpreting, Explaining and Visualizing Deep Learning. Springer, Cham, 2019. 193-209.
12. Böhle, Moritz, et al. "Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification." Frontiers in aging neuroscience 11 (2019): 194.
13. Samek, Wojciech, et al. "Evaluating the visualization of what a deep neural network has learned." IEEE transactions on neural networks and learning systems 28.11 (2016): 2660-2673.
14. Adebayo, Julius, et al. "Sanity checks for saliency maps." Advances in Neural Information Processing Systems. 2018.
15. Sixt, Leon, Maximilian Granz, and Tim Landgraf. "When Explanations Lie: Why Many Modified BP Attributions Fail." arXiv (2019): arXiv-1912.