UPPSALA
UNIVERSITET

PROJECT REPORT

Automated diagnosis of neurodegenerative diseases from PET images of the brain

A proof of concept

Andreas Lycksam, Martin Edin, Nils Erlanson

Project in Computational Science: Report

March 2, 2020

Department of Information Technology

Abstract

Finding unambiguous diagnosis for different Parkinsonian disorders is vitally important, as wrong treatment could have severe side effects. Positron emission tomography (PET) has proven to be a useful method to observe biological processes in the brain. Together with the selective dopamine transporter tracer substance N-(3-iodoprop-2E-enyl)-2β-carbomethoxy-3β-(4-methylphenyl)nortropane (PE2I), labelled with the [^{11}C] isotope, it may provide useful information for diagnosing patients with such disorders. The [^{11}C]-PE2I PET scan, developed by a team at Uppsala University Hospital, provides the necessary information for diagnosing patients with such disorders. Due to the achieved success of convolutional neural network (CNN), an appropriate step was to investigate if a CNN could predict these diseases based on the scans. This project implemented different techniques to explore whether that was possible. Due to the limited amount of data, the main task was to extract more information and aid the network through transfer learning and data augmentation. The best performing method produced a sensitivity of 81%, 75%, 57% and 92% for normal healthy persons, and patients with Vascular Parkinsonism, Parkinson's Disease and Lewy Body Dementia respectively. Important to note is that the data set was not sufficiently large to conclusively make claims about the accuracy on unseen data. A conclusion of the results was that the network seemed to be able to distinguish features that separate the different classes. Consequently, this is reason to further develop this project with additional data and more complex strategies.

Acknowledgements

We would like to thank the team at the Uppsala University Hospital PET-centre for providing us with data along with their genuine interest in the work we have done and Filip Malmberg for the fruitful discussions on approaches and technicalities. Moreover, we would like to express a large gratitude to Maya Neytcheva for an engaging course which further enhanced our competence in both scientific writing and speech.

Keywords— Neurodegenerative Diseases, [^{11}C]-PE2I, Positron Emission Tomography, Machine Learning, Convolutional Neural Networks, Transfer Learning, Data Augmentation

Contents

1 Introduction

Medical imaging techniques have played an important role in understanding the human body by providing information about structures, biological changes and anomalies. Furthermore, medical imaging has been proved to be a contributing factor in gaining knowledge about the central nervous system and provides different techniques to evaluate pathological processes in the brain. In the field of neuroimaging there exists several different types of imaging techniques. The most commonly used are computed tomography (CT) and magnetic resonance imaging (MRI). They are easy to use and provide images with high spatial resolution, mainly used to view structural changes from tumors, bleedings, inflammation, increased intracranial pressure or atrophy. Positron emission tomography (PET) and single photon emission computed tomography (SPECT) have less spatial resolution but can provide physiological, biological and pharmacological information of brain metabolism. To target these functions a molecule involved in a specific process labelled with a positron emitting isotope, forming a tracer, is administered to the patient. A scanner detects the emitted radioactivity and images of the targeted process in the brain can be produced, these images highlight neurological disorders or abnormalities that allude to certain diseases. These images could either be used in tracking disease progression, evaluation of treatment response or ultimately be a decisive input in a diagnose.

During the last decade, a team at the Department of Surgical Science, Radiology, at Uppsala University Hospital has developed a new imaging method to visualize the features necessary to determine unambiguous clinical diagnosis of Parkinson's disease and atypical Parkinsonian disorders. Previously, a dual-scan approach has been used to examine the dopamine transporter availability and the overall brain functional activity in separate examinations of the patient. Instead, a single dynamic PET-scan with the highly selective dopamine transporter PET ligand [^{11}C]-PE2I has shown that it is possible to measure both the amount of available dopamine transporter receptors and the cerebral blood flow (CBF). A single scan approach is simpler and reduces the radiation dose to the patient. Moreover, [^{11}C]-PE2I PET has shown to be more sensitive and specific compared to earlier methods. A pragmatic step in the development process is to evaluate whether the image data produced from this method can be used in some automated software in order to further aid a more robust and reliable diagnosis and/or information of treatment options.

Thus, the aim of this project is to investigate whether and to what extent machine learning methods, more explicitly convolutional neural networks, can be used to automatically provide a correct diagnosis from the [^{11}C]-PE2I PET scans. Specifically, the goal is to classify healthy patients (N) and the common neurodegenerative disorders Parkinson's disease (PD), Vascular Parkinsonism (VP) and Lewy Body Dementia (LBD).

2 Theory

This section provides an overview of the medical processes behind the concerned diseases as well as describing the underlying theory of the chosen algorithms and methods. Finally, a review of current similar research is given.

2.1 PE2I Positron Emission Tomography

The idea of PET is to modify a biological compound such as a peptide, amino acid, protein or antibody involved in a specific physiological process by incorporating short-lived positron emitting isotopes such as [^{11}C] or [^{15}O]. This method is known as the tracer concept. After injection of the tracer, the decay is measured by a highly sensitive ring-structure surrounding the brain that registers gamma-rays emitted from the annihilation of a positron colliding with an electron. Using the fact that the gamma-rays are emitted in opposite directions, the coincident registration results in so called counts. This allows to identify a line along which the annihilation is located. The counts are subsequentially reconstructed whilst also being corrected for physical error sources, resulting in images showing the radioactivity concentration of the measured tissue [4]. This is illustrated in Figure 1.

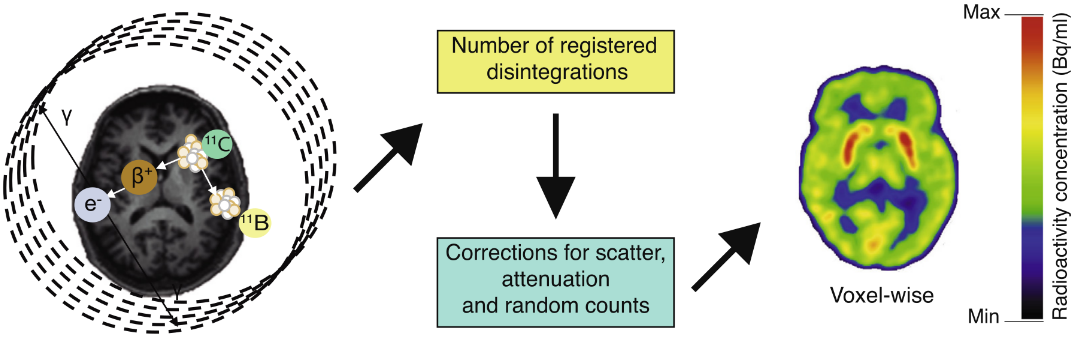

γ
^{11}C
β$^+$
e$^-$
^{11}B
γ
Number of registered disintegrations
Corrections for scatter, attenuation and random counts
Voxel-wise
Max
Min
Radioactivity concentration (Bq/ml)

Figure 1: Schematic illustration of PET-scan process using the tracer [^{11}C]-PE2I. Permission was granted from the authors of [4] to use the illustration found in the published paper.

The radioactive tracer should not exert any pharmacological action toward the targeted area, therefore only a small amount of radioactivity is administered to the patient. Due to the extremely sensitive detectors that register isotope disintergrations at a subpicomolar concentration level, the required amount of administered tracer can be sufficiently low to not cause any pharmacological or blocking effects at the target receptors.

The cocaine derivative PE2I binds highly selectively and specifically to the dopamine active transporter (DAT). The DAT protein is situated in the end of the neuron that produces dopamine and regulates the amount of dopamine in the synapse. It is the same neuron that is affected in Parkinson's disease and thus measuring the amount of DAT proteins gives a measure of the number of viable dopaminergic neurons. Performing PET-scans with the tracer [^{11}C]-PE2I has produced reliable and reproducible DAT estimates. Also, by doing dynamic scans it is possible to measure the CBF in the brain and, hence, track the overall brain functional activity. These combined features can support the discrimination of Parkinsonian disorders and can therefore be a good alternative to the conventional approach of a dual-scan. The dual-scan performs separate scans to investigate both features by using [^{18}F]-FDG PET and [^{123}I]-FP-CIT SPECT imaging methods to provide information about the CBF and DAT availability, respectively.

2.2 Neurodegenerative diseases

Neurodegenerative diseases encapsulate a group of conditions which cause a progressive degeneration and death of neuronal cells. Neurons are the main cells in the brain and spinal cord and damaging these may cause both cognitive and motorical symptoms. As neurons generally do not reproduce or get replaced, the damage done by these diseases is irreversible. In the nearby future there may exist disease modifying therapies, see [1], that can halt progression of the disease, but in order for these to be successful the treatment needs to be initiated at an early stage of the condition. Thus, choosing the right type of treatment early in the progress is essential to achieve an effective symptomatic treatment.

Identifying the correct diagnosis for symptomatically similar diseases is generally difficult, this is clear in the case of Parkinsonism. The diseases in this category exhibit symptoms usually associated with PD, but the underlying cause for each disease varies. Therefore, using the wrong type of treatment could have severe side effects. The prevalent factor for these neurodegenerative diseases is that there is a change in activity and functions in certain areas of the brain. Mainly, the predominant factors to be considered are the overall CBF and the dopamine activity in the striatum, these alone may make it feasible to distinguish distinct patterns between the diseases. The striatum resides within the centre of control for voluntary motor movements. It receives dopamine inputs from various sources which then affect motor movements, a loss of functionality could produce symptoms such as tremors, rigidy and hypokinesia. CBF is a measure of general functionality of different areas in the brain and a subnormal activation in a certain part may inhibit the capability of the corresponding functionality. Understanding the resulting patterns by these two factors of the concerned diseases in this project, i.e. LBD, PD and VP, is helpful to understand which characteristics the neural network will try to detect.

A cause for PD is a decay of dopaminergic innervation in the striatum, meaning that there will be less release of dopamine for patients with this disturbance. Hence, the level of dopamine activation in the striatum is a good indication for PD. On the contrary, VP is the result of small strokes in the brain and thus the striatum is not affected in the same way as with PD. The consequences from the strokes are instead noticed in the CBF. LBD arises from a collection of Lewy bodies in nerve cells, which is a collection of a protein called alpha-synuclein that has numerous important functions for the neurons. Several of the functions are constrained when the proteins are lumped together. The Lewy bodies are spread out through the brain which means that both the CBF and the dopamine levels are subnormal. A healthy patient exhibits a clear activation in the striatum and a higher and more consistent CBF. Figure 2 demonstrates typical characteristics of uptake.

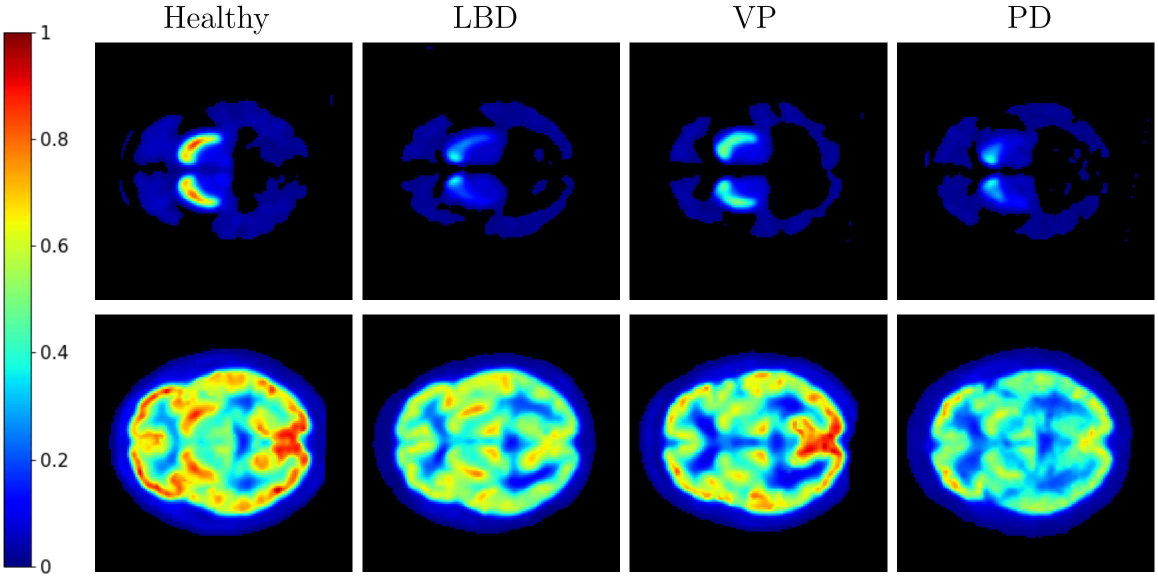


Figure 2: One slice in the brain illustrating the differences of the classes. The scans on the top shows the DAT and the bottom ones shows CBF, the scale is an arbitrary unit of activation.

Looking at Figure 2, the healthy brain shows clear activation in both the CBF and the striatum, whereas the LBD patient has a clear decay in both these factors. The VP scan shows clear activation in the striatum but considerably less blood flow which can be seen by the bigger cavities in the brain, shown in dark blue. Finally, PD exhibits low dopamine levels. It is clear that to get a correct diagnosis both scans are needed, since only looking at the dopamine levels is not enough to separate the classes. Finding characteristic features in the CBF requires a trained eye, but the hope is that a deep neural network will be able to recognize the patterns.

2.3 Machine learning

The goal of machine learning (ML) is to provide a way for systems to automatically learn and improve by experience without explicitly being programmed to do so. This is possible by using data to synthesize useful concepts and structures in order to be able to make future predictions of unseen, related data. There are multiple branches of ML algorithms, however in this project only supervised learning is considered.

Supervised learning has proven to be an effective tool for classification tasks [7]. Classification yields a qualitative output, such as a class, family or type. It could either be binary, which aims to separate two different entities, or multi-class which intend to classify three or more instances. The method uses labeled data in order to learn specific patterns and structures. A key aspect which makes it feasible for the algorithm to learn is the available data. The data is an essential and fundamental factor that allows ML algorithm to be accurate. However, a large amount of data is not by itself sufficient, the data needs to fulfil certain criteria in order to be useful. Consequently, a big part of ML is data preparation which consists of several procedures to reassure that the necessary criteria are met. This section describes techniques related to supervised learning that has been utilized in the project and the underlying algorithm used is described.

2.3.1 Neural Networks

Neural networks are used in a wide range of applications. Their broad applicability originates from the fact that these offer a flexible way of finding and understanding non-linear relationships in given data without the need for prior definitions and hypothesises. The networks are usually composed of three types of layers; input layer, hidden layer and output layer, where each layer consists of a number of neurons. A common practise is to use several hidden layers, constructing a deep neural network. The network forms a sequential construction where each hidden layer is often composed such that it stores different features of the previous layer. In theory, there is no limitation regarding the number of hidden layers. In practice, however, using more layers requires large amounts of memory as the number of weight parameters ranges to the order of millions. The structure of a simple neural network with an input layer, two hidden layers and an output layer is shown in Figure 3.

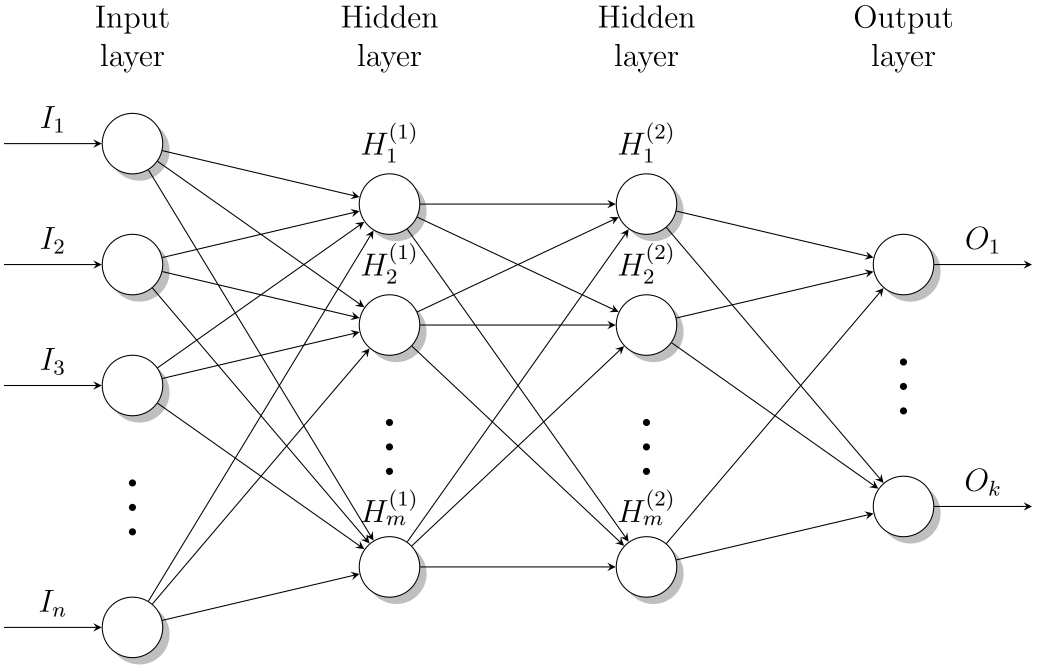


Figure 3: An illustration of a simple neural network with two hidden layers and a multi-class output layer.

Figure 3 illustrates how the input is passed through the hidden layers to the output layer. No computations are made in the input layer, its only purpose is to further pass the information into the network. The neurons in the hidden layers are based on the so-called activation functions, that introduce the non-linearity into the output of each neuron. Each neuron acts as a filter such that the subsequent layers receive values within an acceptable range. The neurons gather the values passed from the previous layer and calculates a weighted sum and applies a bias as follows,

$$H_j^{(l)}(x) = z = \sum_{i=1}^{m}(w_i \cdot x_i) + b_j. \tag{1}$$

The variable l is the number of hidden layers, j indicates one neuron in the current layer, m is the number of neurons in the previous layer, w_i are the weights, x_i is the input values to the neurons and b_j is a bias. After the given activation function is applied on the result z, it may or may not pass the output to the subsequent layer. Due to its simplicity, a common activation function for the hidden layers is the ReLU function, defined in eq.(2). It returns zero if the value is negative or the value itself when it is positive. Each neuron in the output layer represents a class that the algorithm aims to classify. For the output layer, a common activation function regarding multi-class classifications is the softmax function, cf. eq.(2),

$$\begin{array}{cc} \text{Softmax} & \text{ReLU} \\ \sigma(z) = 1/\big(1+e^{-z}\big), & R(z) = max\big(0, z\big). \end{array} \tag{2}$$

Softmax yields a probability for each class, the highest probability is the prediction. This introduces a way of measuring of how certain the model is in its predictions, which is vital information in the training process of the network.

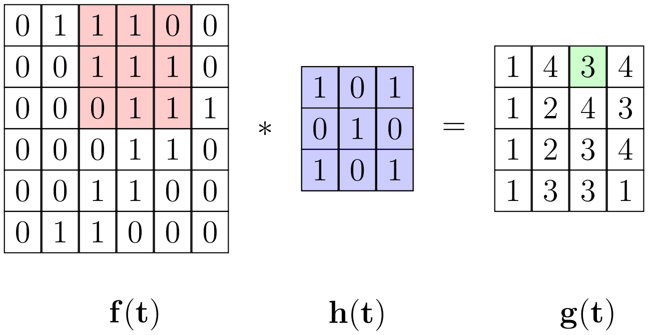
2.3.2 Convolutional Neural Networks

By the use of convolution, the network is able to find more complex patterns in the inputs as it works through the images locally, hence preserving the spatial information. Furthermore, it is substantially more memory effective when it comes to image inputs, as the input size often reduces as it passes through the layers. An implication of this important factor is that it allows for deeper networks and a more elaborate structure.

As the name suggests, these networks are based on the mathematical operation convolution. Convolution is defined as the integral of the product of two functions where one of them has been shifted and reversed and the result expresses how the shape of one of the functions alters the other. The definition in continuous time is

$$g(t) = f(t) * h(t) = \int_{\infty}^{\infty} f(\tau)h(t-\tau)d\tau, \tag{3}$$

where *f(t)* is the input image, *h(t)* is the kernel and *g(t)* is the result. Convolution is conceptually moving the kernel locally with a predefined stride over the input image. An illustrative example of convolution for one channel can be seen in Figure 4, using an arbitrary image and kernel.

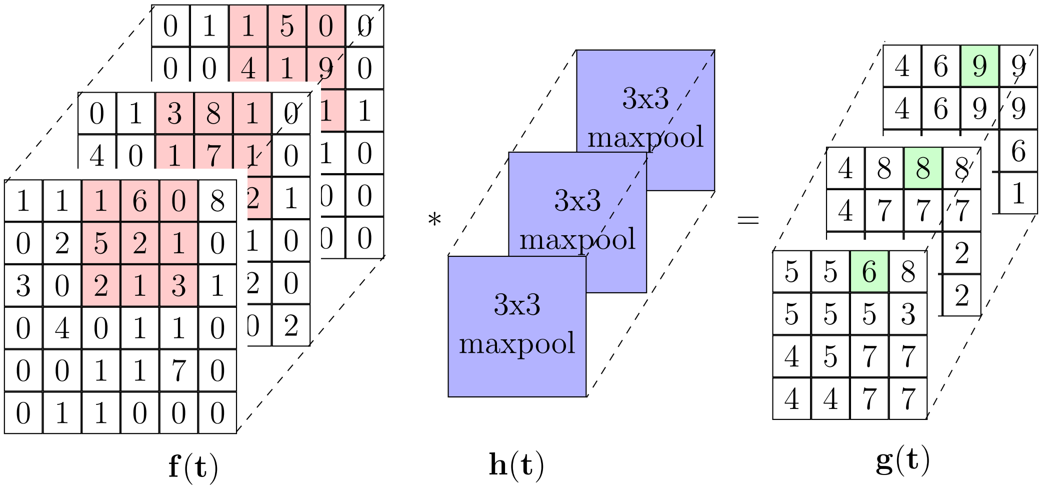


$$\begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 4 & 3 & 4 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 3 & 4 \\ 1 & 3 & 3 & 1 \end{bmatrix}$$

Figure 4: Illustration of a convolution in 2-dimensions using an input f(t) using one channel convolved with a 3x3 kernel h(t) with a stride of one. The result is a feature map g(t).

By altering the kernel in each layer it is possible to extract low-level features such as curves and edges. Combining these features ultimately describes more complex objects and geometries such as faces, buildings or trees. Some common fixed parameter kernels used for image classification are MaxPooling, MinPooling, AvgPooling or MedianPooling, where MaxPooling implies that the maximum intensity is chosen in the pixel neighbourhood. The idea for the others is the same as for the MaxPooling. When the kernel is not fixed, the parameters in the kernel are updated using gradient descent optimization.

CNNs can utilize multiple-channel spatial data in order to process for example different color schemes. The channels should represent the same object with slightly different information and adding these together ultimately expresses the full image. Using multiple-channel input, one kernel per channel is utilized such that each kernel produces its own set of output feature maps. The hidden layers are treated as tensors with dimensions (rows x columns x channels). An example with 3 channels can be seen in Figure 5.



$$\begin{bmatrix} 1 & 1 & 1 & 6 & 0 & 8 \\ 0 & 2 & 5 & 2 & 1 & 0 \\ 3 & 0 & 2 & 1 & 3 & 1 \\ 0 & 4 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 7 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} * \text{3x3 maxpool} = \begin{bmatrix} 5 & 5 & 6 & 8 \\ 5 & 5 & 5 & 3 \\ 4 & 5 & 7 & 7 \\ 4 & 4 & 7 & 7 \end{bmatrix}$$

Figure 5: Illustration of convolution with an input f(t) of multiple channels with a 3x3 Maxpool kernel h(t) with a stride of one. The result is three feature maps g(t), one for each channel.

A three-dimensional CNN is not much different from the previous example in Figure 5. The difference can conceptually be seen as the kernel moving in the third orthogonal direction. Instead of considering pixels, the elements are voxels. By considering volumes the requirements regarding memory increase as the number of parameters in the system increases enormously. Consequently, the networks usually only consider small 3D patches of the volumes to be examined, by removing redundant information or similar techniques. Using multiple channels is still possible, it simply adds a dimension to the tensor.

2.3.3 Training the network

The network's parameters are obtained by a process, referred to as training. The training is an essential step for the network to find the important features for each class. A cost function is introduced, which quantifies the error of the network's prediction compared to the ground truth and statistical tools are used in order to measure this performance. This cost function together with optimization, which updates the parameters in order to improve the performance, are the two main concepts when training the model. Which cost function to be used depends on the problem formulation, in a multi-class problem a common cost function is the *Cross-Entropy loss*. In an discrete case, it is defined as

$$L_{CE}(x,t,\theta) = -\sum_{i=1}^{C} t_i log\big(p(x,\theta)\big), \tag{4}$$

where C is the number of classes considered, i denotes the current class considered, t is the ground truth transformed into one-hot-encoded label format, i.e. a zero or a one, and p is the prediction score output from the model according to the input x and the current parameters θ. The measure is also applied on the test data, which gives an indication of the performance on unseen data. A test loss gives an indication on how general the model is and how it will perform on unseen data. With the cost function we state the following optimization problem,

$$\hat{\theta} = \arg\min_{\theta} J(\theta), \qquad J(\theta) = -\frac{1}{n}\sum_{j=1}^{n}\sum_{i=1}^{C} t_i log(p(x_j,\theta)), \tag{5}$$

where n is the number of data points in the set and $\hat{\theta}$ is the optimal parameter values. In each step of the optimization, the parameters are updated to reduce the cost function until a local optimum is reached. The procedure works as follows,

$$\theta_{k+1} = \theta_k - \gamma\nabla_\theta J(\theta_k), \qquad \nabla_\theta J(\theta_k) = \frac{1}{n}\sum_{j=1}^{n}\nabla_\theta L_{CE}(x_j,t,\theta_k), \tag{6}$$

where the γ is the learning rate of the model. The training process requires considerably high amount of computations. This occurs both due to the size of the training data and the amount of parameters in the network. A combination of two different methods are often used to overcome the problem, stochastic gradient descent(SGD) and back propagation. SGD is a method that trains on a batch of the data instead of having to sum over the entire set, which consequently decreases the needed computations to the size of the chosen batch. For this to work it is necessary for the batch to be representative of the entire set. Back-propagation eases the computational load by applying the chain rule of calculus in order to find the partial derivatives of the loss function. The load becomes lighter as the gradient calculation does not solely rely on differentiation approximations.

2.3.4 Validation

A vital part in ML problems is to have a tool to measure the performance. The tool is used as a comparison to other models and for finding the best configuration. It is not enough advisable to analyse the performance of the model based solely on the training data. Instead, the true performance is evaluated on unseen data. Thus, a split of the data is done into a test and training set. The goal of the split is to have a test set that is representative of the entire data, while maximizing the size of the training set.

The k-fold cross-validation is a common method that both validates the entire data set and describes whether the test set is representative. The idea is to split the data set in to k-folds. Choosing one fold as the test set, the training is done on the remaining data. This procedure iterates through the entire set, altering the test and the training set.

The performance of each fold is then averaged out and the result indicates a level of robustness, since the model has been tested on all data. Disadvantages with the method are that it becomes computationally heavy, as it needs to train the model k times.

2.3.5 Data preparation and augmentation

Data preparation starts at the data acquisition, which needs to be standardized to reproduce the same scenery regardless of what is to be measured or captured. In medical imaging, there are several factors that produce variances in the image caption procedure, unrelated to the classes. Variations may originate from differences in the machinery and methodology as well as individual factors such as geometry of the skull, the actual uptake of the tracer and that the head may lie in different positions while the scans are performed. These factors needs to be addressed and this procedure is usually called standardization. Hence, numerous steps are performed before training of the model. The patient's head is fitted into a standardized template, rotated and shifted regarding all the spatial axes in order to align the brain in all directions. The individual uptake ratio is dealt with using a method called *Standardized Uptake Value-1* which states that the injected radioactivity is the same as the whole body concentration during the scan, additionally the intensity levels are normalized to the cerebellum in the back of the head.

All these steps ensure that a lot of variance in the data is removed, which is helpful for the machine learning algorithms to work efficiently. This is especially true for classification problems when the data set is small, since the intrinsic variance of each sample will represent an disproportionate large part. This introduces difficulties for the model to distinguish the intrinsic variance from the variances separating the classes. Thus, an important step before training is to inspect the data and get some undertanding of how the classes vary and how the individual samples within the classes differ. After inspection, the main objective is to minimize the intrinsic variances while maintaining the class differences. Thereafter, producing new samples with the remaining intrinsic variance is done to accomplish a more robust model as it becomes more prepared for these variations. This is called data augmentation. Extremely crucial for using data augmentations is that the data needs to vary by itself in the introduced augmentation, otherwise the model will be trained on variances that do not exist in the test and validation set.

Normalization means to scale the intensities into a given range. The importance of this step differs from application to application. Generally, to require values between zero and one is a common technique regarding neural networks algorithms, however the actual normalization method depends on the problem formulation. If the values are relative to each other, that relation should be preserved within the new range. It is then important to separate the training normalization and the test normalization, as the test should be normalized relative to the training set. If there is no need to preserve relative values between data points it is often valuable to normalize each sample to itself. The normalization of the test set is thereafter not dependent on how each training set is normalized and is i general more straightforward.

2.3.6 Transfer Learning

The achieved success of CNNs is partly due to the magnitude of annotated images, such as the ImageNet from Stanford University. Consequently, there has been various investigations of whether a neural network trained on such data sets can be transferred and used on unrelated data. This idea has shown to be highly promising (cf. [14]). As it can reduce both computational needs and increase performance,(cf.[14]).

The main idea is that the initial layers of CNNs have similar filters, that pick out high level features, which then proceed to the next layers that get increasingly detailed. Thus, regardless of the classification task, using a pretrained network seems to increase the performance compared to training the network from scratch. This is especially valuable for problems with small data sets, as the network can be forced to focus on the more detailed features associated with the task. By freezing the early layers and only applying the gradient descent on the later layers and the fully connected layer, ensures that the network only updates the parameters in the later layers.

A popular CNN architecture for image recognition is ResNet, first proposed in ”Deep Residual Learning for Image Recognition” [3]. These networks uses the so-called *Residual blocks* in attempt to increase the depth of the model, as research articles have shown great promise for the correlation between depth and accuracy [11]. In naive attempts to build deeper networks, i.e. simply adding more layers, the training error increases after a certain depth [3]. Residual blocks present a solution to this problem. The output from a block is an addition of the actual outcome from the previous layer and the input from two layers earlier, called Identity mapping. These two components are then projected to the same size in order to combine them. This procedure is described in Figure 6.

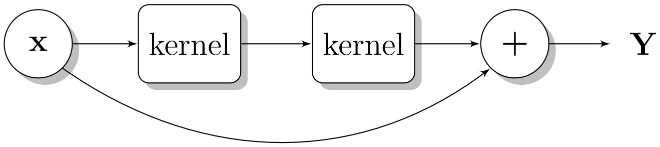


Figure 6: Block scheme describing a Residual block. **X** is the input in the residual block and is the output from an arbitrary layer in the network. **Y** is output and is calculated as a sum from **X** and the outputs from the two previous layers.

2.4 Previous work

Machine learning practices for pattern recognition in medical imaging are quite common, however only a small amount of the work considers a multi-class classification of diseases. Most of the articles only regard binary classification, i.e. to separate healthy and normal against disordered and abnormal using alternative methods for deep learning due to data acquisition concerns.

The work in [2] presents a variation of a multi-class relevance vector machine classification method using FDG PET-scans. It is able to differentiate between PD and three different types of atypical parkinson's disease, in particular multiple system atrophy (MSA), progressive supranuclear palsy (PSP) and corticobasal syndrome (CBS). The method is tested on 120 patients and the data used was only cerebral FDG uptake pattern images. Conclusively, this method fails when considering differentiation and separation of the atypical Parkinson's diseases. Another study shows a similar application and input data but utilizes a deep learning approach [13]. The volume data used is compressed into 2D images using tensor-factorization. Furthermore, the CNN implemented is pretrained on a large data base of FDG PET-images. The study is based on 257 patients and the approach successfully demonstrated a sensitivity measure above 87% for MSA, PSP and PD. Another suggestion to tackle the classification of PD, a normal control group (NC) and Scan Without Evidence of Dopaminergic Deficit (SWEDD) uses MRI and diffusion-weighted tensor imaging (DTI) images of the brain as baseline data [8]. A distinction from other reviewed methods is that additional information such as three cerebrospinal fluid biomarkers and clinical scores including depression scores, sleep scores, olfaction scores and a certain cognitive assessment scores is utilized to boost the performance of the model. The work is based on 208 patients and the model is implemented using a support vector machine where it is clearly stated that the supplementary data fed into the system yielded positive benefits in the accuracy of the model. Another work that uses MRI to identify subtle anomalies present in the brain coupled with PD uses a CNN [10]. The authors claim that their method is able to locate the most discerning regions on the MRI images in order to be successful in the separation of PD versus atypical parkinsonian disorders.

Auxiliary research regarding classifying other neurodegenerative diseases, such as Alzheimer's disease (AD), using machine learning is helpful to further gain perspective on our work. A thorough investigation of binary classification between AD and NC uses a data set consisting of the MRI scans of 1455 patients [12]. In the study the authors compare the performance of 3D and

2D models. The 3D models, presented there, are all pretrained on the entire training set using autoencoders and the 2D model uses a *ResNet* architecture pretrained on ImageNet, with redundant slices removed from the input. The achieved results indicate a higher accuracy for the 3D approaches. A similar work, based on 3D sparse autoencoders, extracts generic features coupled with AD [5]. Additionally, the authors deploy transfer learning as well as deep supervision on the latter layers. The method is able to classify AD, mild cognitive impairment and NC with an accuracy of above 75%.

3 Method

In order to produce a usable algorithm, there is a need to create a pipeline which streamlines all the steps of the project. The methodology of the project starts at the data acquisition and ends at the classification. The workflow of the project is depicted in Figure 7 below.

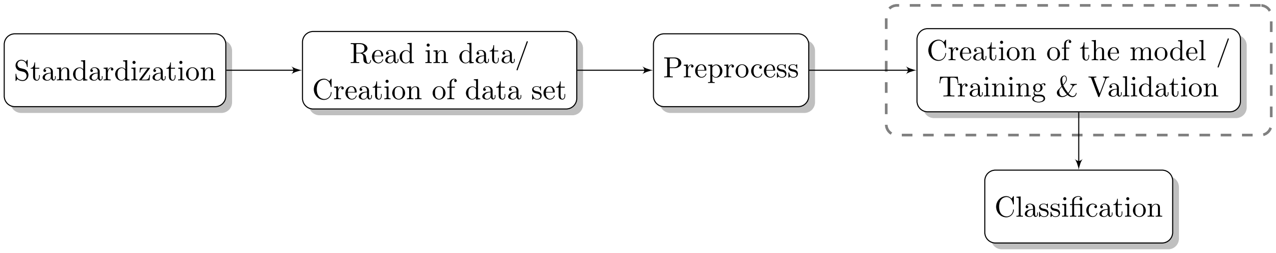


Figure 7: Block scheme of the approach of the project. The dotted lines indicates that the training and validation is only done once to create the model with the original data set, which then makes it possible to classify new samples from the trained model.

The framework used to implement the algorithms used is PyTorch and the data used in this project are provided by the PET-centre acquired with their new PE2I method. The data set includes 20 labeled patients, divided in four classes - four PD, four VP, five LBD and seven healthy brains. The data of each patient consists of two scans, one DAT and one CBF scan, of format 128x128x128 voxels. From this, two approaches are possible, either a model which uses a volume or a single slice as inputs. This project mainly focuses on using the 2D approach. The first step is to standardize the data, that is to spatially align all scans to the same reference coordinate. This has been done by the people at the PET-centre. Thereafter, to make the data compatible with the ML framework, the loaded scans are structured such that each patient's sample are grouped together as two channels, representing both features from the scans. The first component represents the scans, represented as a 4D tensor, and the other component in a sample the corresponds to the diagnosed label of the patient. After the data set has been split, several preprocessing methods were applied. At this step the project diverged into two branches, 3D volume and 2D slices, which handled the succeeding steps differently. The following sections describes the two approaches.

3.1 2D models

The idea is that a carefully chosen slice in the vital area of the brain contains significant information to distinguish the classes. To preserve the relation between the patients, each slice was normalized to values between zero and one with regards to the training set. Thus, the process produced a normalization function that maps the test data into the same range as the normalized training set. After inspection of the data, it was noticed that there were similarities in the behaviour of the activity and intensities between adjacent slices in the brain for each respective sample. Naturally, an augmentation of the data was introduced by representing slices above and below the most significant slice as new samples. Furthermore, as structural symmetry and functional asymmetry is present in the human brain [6] and that the diseases are not contained to a specific hemisphere, another augmentation was to mirror the images along the longitudinal fissure. Thereafter, two types of methods were considered.

The first method was based on performing basic arithmetic on the image data with the endeavour to further nourish the network with additional information. As the prevalent behaviour of the healthy images was a high uptake of the dopamine levels and similar activity in the blood flow, a reasonable mean brain of these could be calculated. This was done by taking the mean of all the normal patients in the training set. The mean brain was subtracted from the images and added into the model as two more channels. Illustrations of this is shown in Figure 8, 9 and the resulting model with four input channels is illustrated in Figure 10.

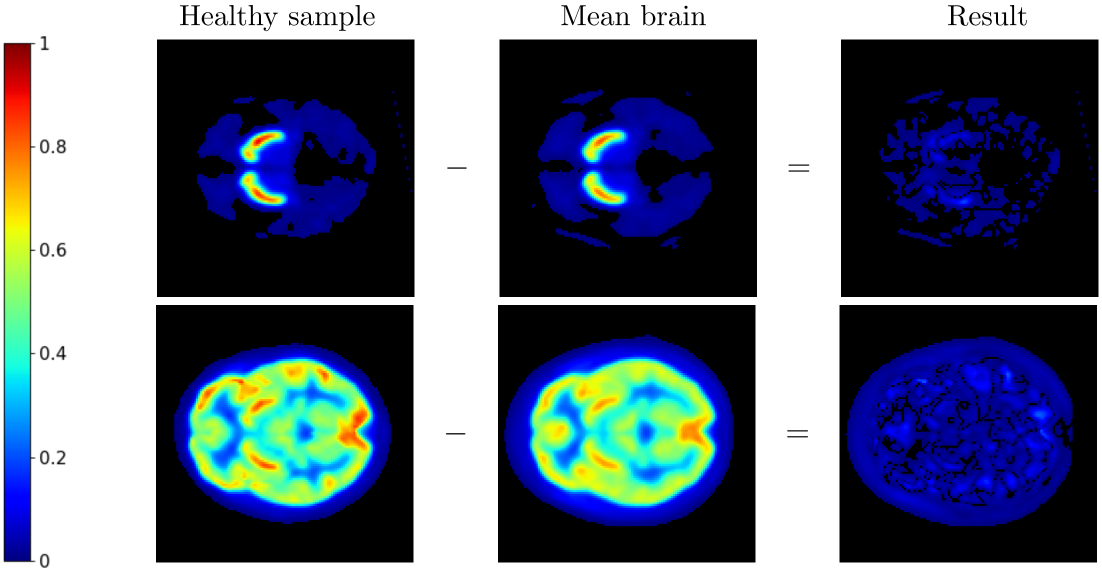


Figure 8: Illustration of the result obtained from subtracting the mean normal brain from a healthy sample.

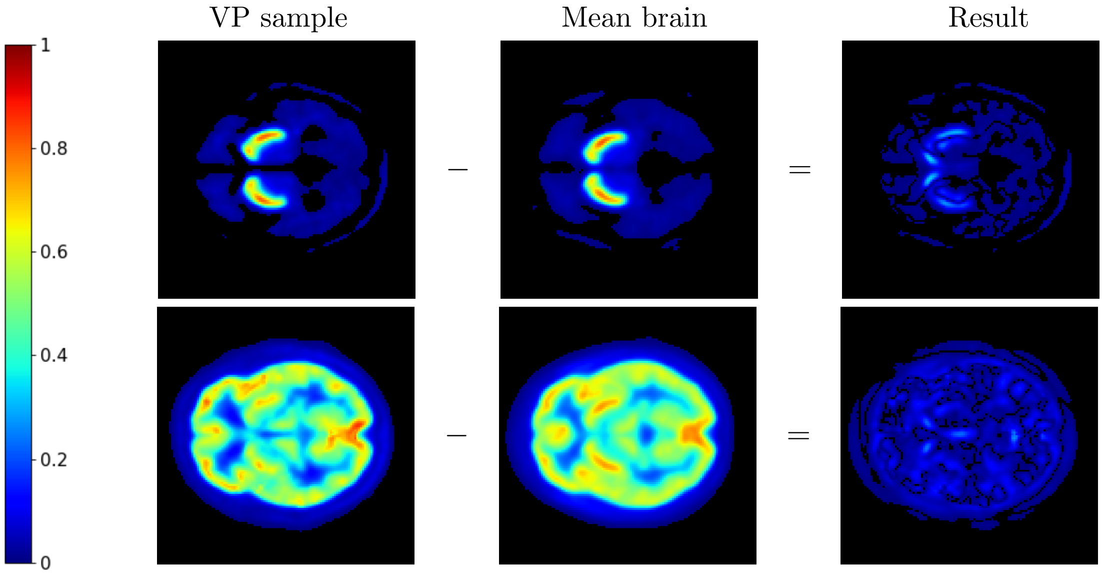


Figure 9: Illustration of the result obtained from subtracting the mean normal brain from a VP sample.

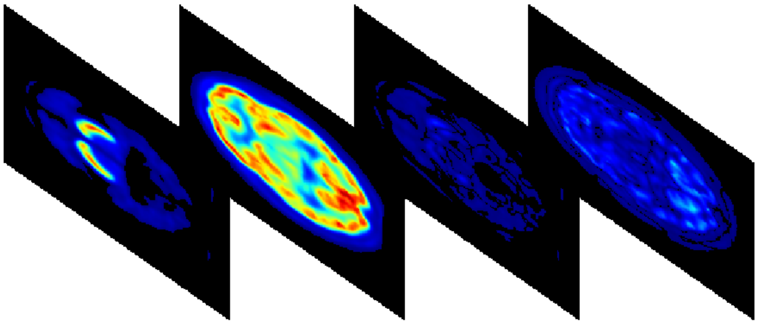


Figure 10: Visualization of the input with four channels. The first channel is DAT availability, the second one is standard CBF, the third and fourth is the result of the subtraction between the standard scans and the mean healthy brain calculated.

The second method arose from the idea of including spatial information from the volumetric data while still using a 2D model. By stacking multiple slices around the most significant slice into one image, the data could be fed into the network as a 2D input. The reasoning behind that this might work was that the activity may change in different ways when considering different diseases and heights in the brain. Figure 11 shows a sample where multiple slices has been stacked.

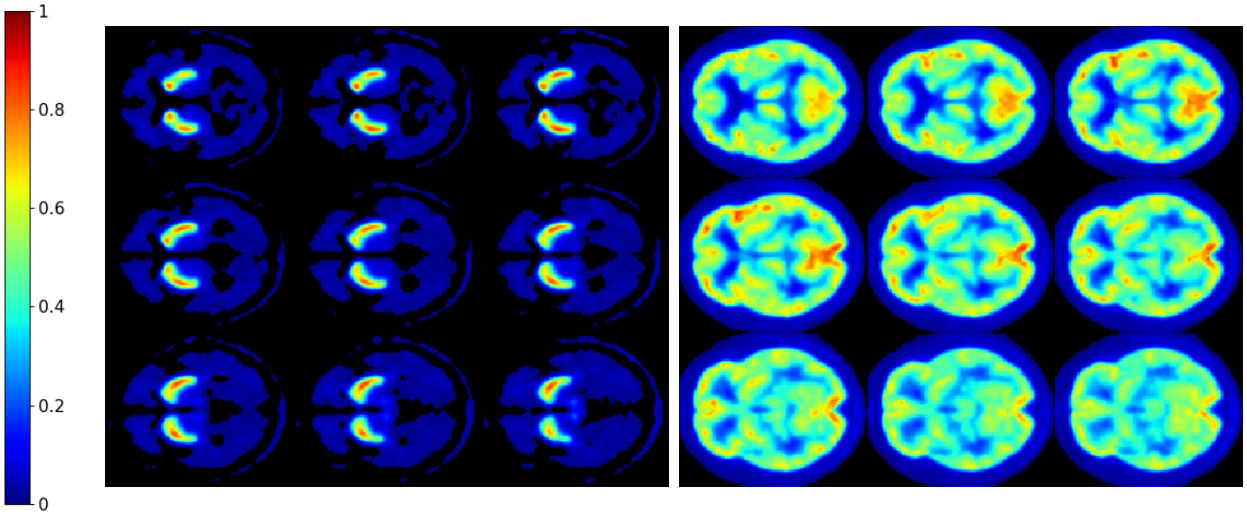


Figure 11: Illustration of the two channels where multiple slices has been stacked around the most significant slice into a 2D image. The slices are placed in ascending order where the top left is the farthest below the central slice, and correspondingly for the bottom right.

A major advantage of the 2D approach is that there is a multitude of pretrained models available. Thus, the sole candidate was PyTorch's 152 layered *ResNet* as the network's architecture. The training mainly occurred at the fully connected layer at the end of the network, where a softmax activation function was used to yield a probability for each class in the prediction.

3.2 3D model

After inspection of the data it was found that a lot of irrelevant and unnecessary information was incorporated in the scans. More specifically, they included glands around the jaw which showed a relatively high uptake and activity as well as superfluous padding between the edges of the image to the actual skull. An important step was hence to crop the volumes such that only the highly relevant information was conserved. This step accomplishes two things, the network will not be mislead on insignificant data and the amount of parameters needed in the model is drastically reduced. Thereafter, to preserve the relation between the patients intensity values each patient was normalized according to the largest intensity value found in the data set.

The architecture of the considered 3D model was heavily inspired by [9]. However, a major part of their success of the pattern classification system described in the paper was the combination of sparse autoencoders and the CNN studied. The sparse autoencoders was not assessed in this project due to multiple reasons, the small amount of data at hand and the fact that due to the complexity of that subject it would require a lot of time and expertise to obtain a useful result.

3.3 Model validation

In order to get a meaningful way to compare different models, augmentations and configurations within the model a specially crafted validation scheme was implemented. Due to the limited data set there is a high probability that the test set contains a majority of the samples of a certain label, resulting in a lack of training samples. To achieve an unbiased training model, the implemented scheme is a variation of k-fold cross-validation where the test set is constructed to contain only one sample of each class. Hence, the data in each fold divided into a 80% training and 20% test data split. As the training data were heavily unbalanced, a weighted sampler was implemented to compensate for the unequal class selection probabilities during the training. Its main purpose is to prevent bias while training. From the cross-validation an accuracy measure is calculated, which includes robustness of the model as it is determined from the predictions done by models trained on different training sets.

Another way to distinguish the models was to look at the behaviour of the training and test loss as these tend to disclose how confident the is in its predictions. The cost function used for the training and validation was cross-entropy loss and the optimizer used to minimize the error was the adaptive movement estimation algorithm, commonly known as Adam.

4 Results

Here we present the result of the three different model approaches with different settings. An important note is that the results presented here are all given as average values of trained models with different permutations of the test/training set and not on an actual validation set. Complementary information about how the models perform is shown by illustrating the training process on in different scenarios. In this section, the best model is referred to as the configuration with the sensitivity highlighted in green and the worse model is correspondingly the sensitivity highlighted in red, shown in Table 1. The measures used in the following table is defined by the following equations,

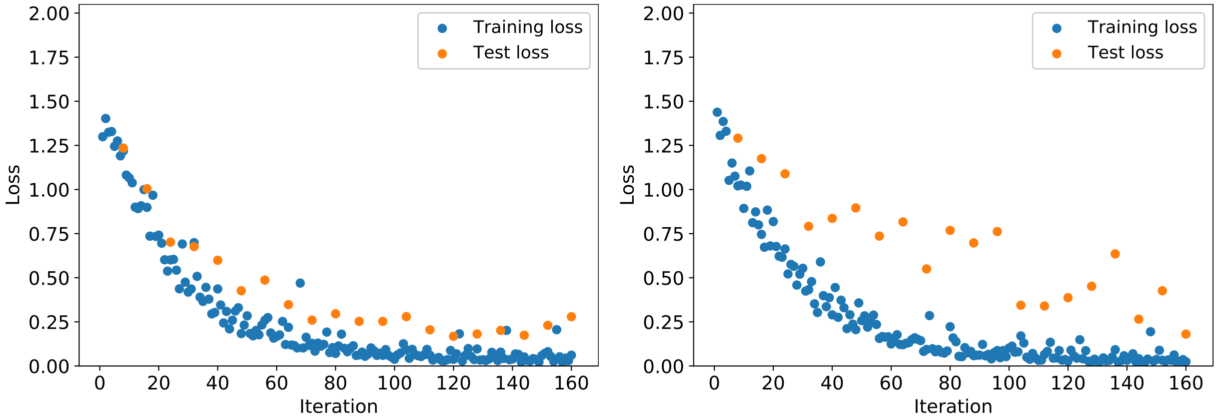
$$\begin{aligned}\text{Predicted Positive Value} &= \frac{\text{TP}}{\text{TP}+\text{FP}},\\[2ex] \text{SENsitivity} &= \frac{\text{TP}}{\text{TP}+\text{FN}},\\[2ex] \text{F1-score} &= \frac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}},\end{aligned} \tag{7}$$

where TP stands for true positive, FP for false positive and FN for false negatives.

Table 1: Summary of the results calculated from 10 5-fold cross-validations from the different methods examined. Where an **x** indicates that the method was used. The validation measure is an average of all the classes' sensitivity.

Classification	Subtraction	Transfer	Data	Validation
method	mean brain	learning	augmentation	sensitivity [%]
2D with one slice	**x**	**x**	**x**	78.0 ± 6.4
		x	**x**	69.5 ± 6.4
	x		**x**	48.0 ± 6.0
	x	**x**		54.5 ± 13.1
				56.0 ± 6.4
2D with multiple slices		**x**	**x**	76.0 ± 5.4
			x	59.0 ± 2.0
				59.0 ± 5.8
3D				40.0 ± 0.0

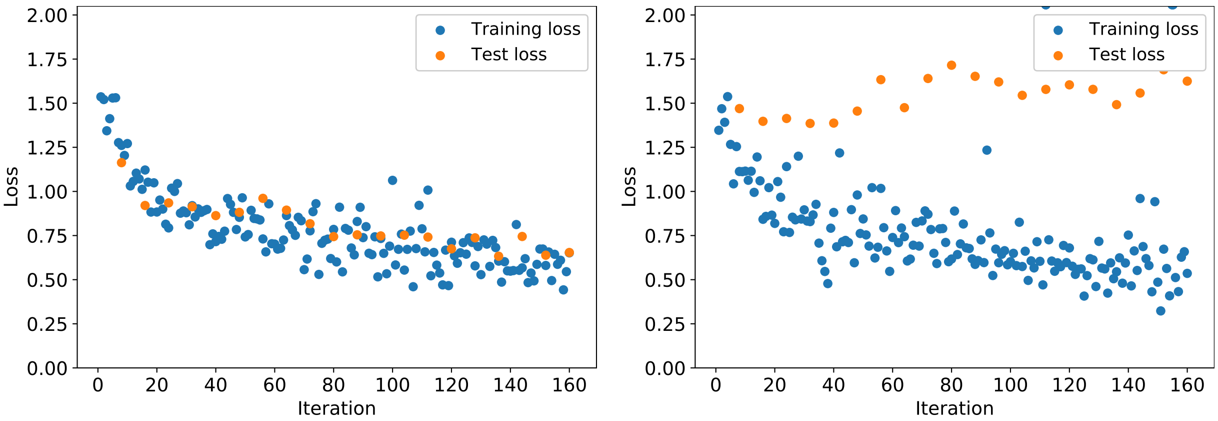
Figures 12 and 13 highlight in more detail the differences between the best model and the worst model, in particular with regards to how well they handle outlying data.



(a) Test set with class representative samples

(b) Test set with abnormal samples

Figure 12: Illustration of the learning procedure on the best model using a test set consisting of samples that resembles the disease classes and one with a test set consisting of samples that has a lot of intrinsic variance.



(a) Test set with class representative samples

(b) Test set with abnormal samples

Figure 13: Illustration of the learning procedure on the worst model using a test set consisting of samples that resembles the disease classes and one with a test set consisting of samples that has a lot of intrinsic variance.

Further investigating the best performing model, highlighted in green in Table 1, the individual classification rates based on eq. (7) for each class is shown in Table 2.

Table 2: Summary of the measures seen in eq. (7), calculated from 10 5-fold cross-validations using the best model approach.

Disease	PPV[%]	SEN[%]	F1-score[%]
N	98.6 ± 4.3	81.4 ± 6.5	90.0 ± 4.2
VP	67.8 ± 8.3	75.0 ± 11.2	70.6 ± 7.1
PD	73.2 ± 20.0	57.5 ± 11.5	63.6 ± 13.6
LBD	72.5 ± 9.1	92.0 ± 13.3	80.6 ± 9.6

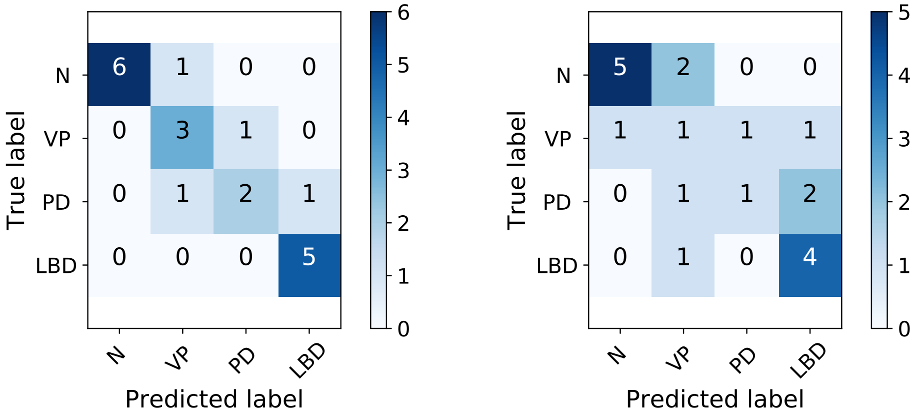


Figure 14: Summary of the performance in forms of confusion matrices calculated from the 5-fold cross-validation based on all of the 20 data samples. The confusion matrix corresponding to the best model is to the left and the worst model is to the right.

5 Discussion

This project faced a large problem due to the limited data set consisting of scans from 20 patients, which is a common problem when performing a ML approach on medical imaging. Having a small data set leads to a lot of complications that needs to be addressed, resulting in the ordinary ML strategies and approaches must be tailored to the specific task at hand. As scans were volumetric, there were possibilities to utilize the information in multiple ways. Using the fact that the most significant information could be extracted from a certain height of the brain located in the center of the striatum, the data was conclusively more useful when it was represented in other ways than in its original format. The study shows that there are several benefits of considering 2D approaches over 3D models in this specific problem. Primarily, there are more available and accessible techniques regarding insubstantial data such as data augmentation and transfer learning when considering 2D model inputs. This was the main reason for the focus on 2D. As mentioned in Section 2.4, a valid method for the 3D model seems to be based on autoencoders. To implement such is out of the scope for this project, therefore the comparison between the 2D and 3D model is not quite fair. However, for the autoencoder approach to work the data set is required to be much larger, therefore there might be a threshold regarding the data set size that would make the 3D approach superior.

Another problem with having insufficient amount of data, is the fact that the intrinsic variances have to be adequately small enough compared to the class variances in order for the model be able to separate these two types of variances. This issue is shown in Figure 12b and 13b, namely, the model is learning patterns from samples which do not fully resemble the outliers which makes the model uncertain in its predictions. It can further be explained that there were outliers in each class which, we believe, inherits characteristics coinciding with another class label. Determining a diagnosis during clinical practise, it is likely that additional inputs were used rather than the scans alone. Additionally, by comparing Figure 12 and 13 the value of using the demonstrated improvements is clear. The best model is able to capture these variances to a greater extent, illustrated by the decreasing test loss, whereas the worst model evidently does not. Further evidence of the improvements is seen in Figure 14. It is noticeable that it is easier to classify the healthy patients and LBD regardless of the implemented model, which explains the intrinsically high accuracy rate as these two classes are often correctly predicted. It further shows that VP and PD has similar attributes to the other diseases, which makes them harder to separate. The differentiation of the best model and the worse is thus determined by the ambiguous samples. This is further clarified in the thorough investigation from Table 2 which shows the score based on the performance measures, eq. (7), for each individual class using the best model approach.

There is also a reason to be sceptical about the performance measures, meaning that the presented results might not be representative of how well the model would perform on unseen data. The reason for this is once again that the amount of data is inadequate, thus making it hard to give a valid claim beyond the data set. As discussed earlier there were some specific samples within the classes that were consistently hard to classify. With the information given, it is impossible to know whether these are actual outliers within the diseases or the classes will always vary to that extent. However, when the data is representative, an automated diagnoses using the [^{11}C]-PE2I-scans is shown to have great promise. With the scans, highlighting different characteristics, the network seemed to be able to gather the necessary features in order to distinguish the conditions.

Finally, this report has focused on displaying what the ML algorithm tries to do and how it works. In the end, it is essential for domain experts and doctors to understand the rationale behind the model's predictions in order to trust it. Without this reliance, an automated solution does not fill any purpose due to the severe consequences of a faulty diagnosis. Clear communication and transparency in the report and between the fields are crucial as there is a lot of ethics concerning these approaches which needs to be addressed in application areas like medicine.

6 Conclusions

It has been shown that an automated diagnosis using a CNN approach with the [^{11}C]-PE2I PET scans is promising. In this project, utilizing the volumetric data in a 2D fashion generated the best results as it made it feasible to exploit accessible ML techniques. However, the results may be inconclusive to fully reject a 3D model approach as the limited data did not support this methodology. In the end, it is important to note that the results obtained may not disclose the behaviour on fully unseen data as it is not statistically significant to makes these claims.

References

[1] J. Cummings. Disease modification and neuroprotection in neurodegenerative disorders. *Translational neurodegeneration*, 6(1):p: 25, 2017.

[2] Gaëtan Garraux, Christophe Phillips, Jessica Schrouff, Alexandre Kreisler, Christian Lemaire, Christian Degueldre, Christian Delcour, Roland Hustinx, André Luxen, Alain Destée, et al. Multiclass classification of fdg pet scans for the distinction between parkinson's disease and atypical parkinsonian syndromes. *NeuroImage: Clinical*, 2:883–893, 2013.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[4] Kerstin Heurling, Antoine Leuzy, My Jonasson, Andreas Frick, Eduardo R Zimmer, Agneta Nordberg, and Mark Lubberink. Quantitative positron emission tomography in brain research. *Brain research*, 1670:220–234, 2017.

[5] Ehsan Hosseini-Asl, Georgy Gimel'farb, and Ayman El-Baz. Alzheimer's disease diagnostics by a deeply supervised adaptable 3d convolutional network. *arXiv preprint arXiv:1607.00556*, 2016.

[6] Kenneth Hugdahl. Symmetry and asymmetry in the human brain. *European Review*, 13(S2):119–133, 2005.

[7] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. Supervised machine learning: A review of classification techniques. *Emerging artificial intelligence applications in computer engineering*, 160:3–24, 2007.

[8] Haijun Lei, Yujia Zhao, Yuting Wen, Qiuming Luo, Ye Cai, Gang Liu, and Baiying Lei. Sparse feature learning for multi-class parkinson's disease classification. *Technology and Health Care*, 26(S1):193–203, 2018.

[9] Adrien Payan and Giovanni Montana. Predicting alzheimer's disease: a neuroimaging study with 3d convolutional neural networks. *arXiv preprint arXiv:1502.02506*, 2015.

[10] Sumeet Shinde, Shweta Prasad, Yash Saboo, Rishabh Kaushick, Jitender Saini, Pramod Kumar Pal, and Madhura Ingalhalikar. Predictive markers for parkinson's disease using deep neural nets on neuromelanin sensitive mri. *NeuroImage: Clinical*, 22:101748, 2019.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[12] Junhao Wen, Elina Thibeau-Sutre, Jorge Samper-Gonzalez, Alexandre Routier, Simona Bottani, Stanley Durrleman, Ninon Burgos, and Olivier Colliot. Convolutional neural networks for classification of alzheimer's disease: Overview and reproducible evaluation. *arXiv preprint arXiv:1904.07773*, 2019.

[13] Ping Wu, Abhijit Guha Roy, Igor Yakushev, Rui Li, Sailesh Conjeti, Sibylle Ziegler, Jian Wang, Stefan Forster, Nassir Navab, Markus Schwaiger, et al. Deep learning on 18f-fdg pet imaging for differential diagnosis of parkinsonian syndromes. *Journal of Nuclear Medicine*, 59(supplement 1):624–624, 2018.

[14] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *arXiv e-prints*, page arXiv:1411.1792, Nov 2014.