Parkinson's Disease Prediction by analyzing Micrographia

PROBLEM STATEMENT

Distinguishing different stages of Parkinson's Disease using geometric drawings (spirals and sign waves) using composite Index of speed and Pen-Pressure of Sketching.

OVERVIEW

Parkinson's disease is a progressive neurological disorder where the coordinated bodily muscle movements influenced by a substance in the brain called dopamine's levels deteriorate. These can have a noticeable effect on the handwriting and sketching (drawing)of a person suffering from early stages of the disease.
Here we will analyze the raw images collected in the study and see if we can create a classifier for a patient having Parkinson's and draw some conclusions along the way.

TEAM MEMBERS:

Hari Sindhu
Email Address:hari21110158@snuchennai.edu.in
Keerthana G
Email Address:keerthana21110269@snuchennai.edu.in

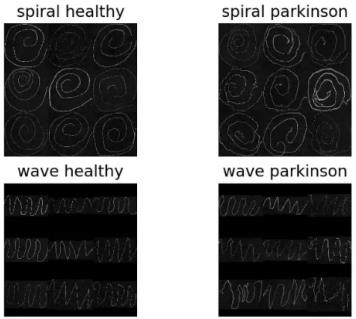
SHIV NADAR UNIVERSITY
CHENNAI

PD Prediction by analyzing Micrographia

LITERATURE REVIEW & ALGORITHM

(i) Image data of the healthy and the patients with Parkinsons drawing spirals and waves are collected.
(ii) The images are further divided into training and testing groups for comparing the results.
(iii) Perform basic image processing by
- Thresholding and cleaning the data
- Quantifying the thickness of the data obtained through nearest neighbours
- Skeletonization
- Intersections and edge points are calculated - a perfect curve has no intersection
(iv) After obtaining some potentially useful metrics that help us not only to understand the degree of micrographia but also can be used as inputs into a classifier such as Logistic Regression or Random Forest.
(v)We will finally perform classification using these metrics and also compare to a more powerful but black-box, neural network. The advantage of using Random Forest is we will also be able to see which features provide the strongest impact to the model.

RESULT SAMPLE



What is Parkinson's Disease???

Parkinson's disease is a long-term degenerative idiopathic, progressive disorder that affects the nervous system and the parts of the body controlled by the nerves.
This is marked by tremors, muscular rigidity and slow imprecise movements.
It causes slow motor movements, depression, anxiety, sleep and sensory system disorders and other behavioral changes.
Researches show that the drawing speed was slower and the pen pressure lower among Parkinson's patients — this was especially pronounced for patients with a more acute/advanced forms of the disease.
Spiral drawing is a skilled and complex coordinated motor activity. Therefore, it is treated as a sensitive motor assessment and a preliminary test for early symptoms of Parkinson's disease.

The main idea behind the implementation is to classify a person as Healthy or having Parkinson’s disease by looking at the Spiral Drawing made by the person. The Spiral Drawing created by a healthy person will look almost similar to a standard spiral shape. However, a spiral drawn by a person with Parkinson’s disease will highly deviate from a perfect spiral shape and look distorted due to slow motor movements and decreased coordination between hand and brain.

The five stages of Parkinson

Stage 1

Mild symptoms that do not typically interfere with daily life, including tremors and movement issues on only one side of the body.

Stage 2

Symptoms continue to become worse with both tremors and rigidity now affecting both sides of the body. Daily tasks become challenging.

Stage 3

Loss of balance and movements with falls becoming frequent and common. The patient is still capable of (typically) living independently.

Stage 4

Symptoms become severe and constraining. The patient is unable to live alone and requires help to perform daily activities.

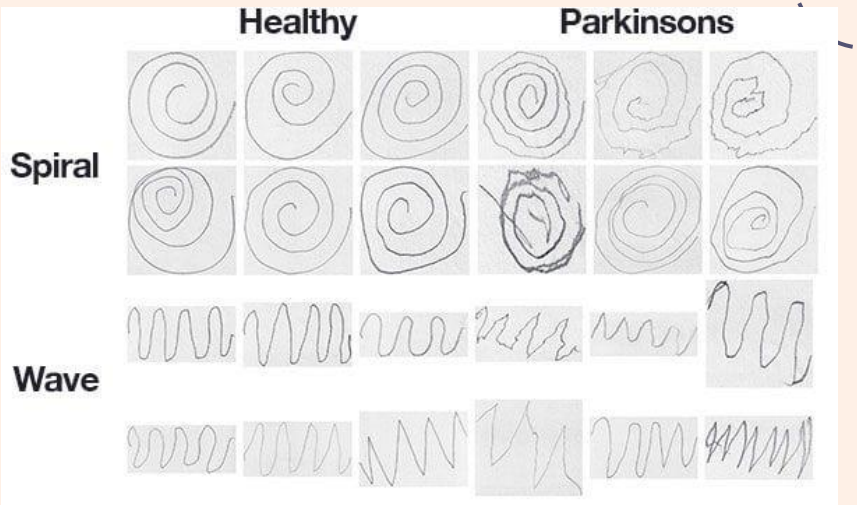
Stage 5

Likely impossible to walk or stand. The patient is most likely wheelchair bound and may even experience hallucinations.

About the data

The dataset consists of 204 images and is pre-split into a training set and a testing set, consisting of:

- **SPIRAL:** 102 images, 72 training, and 30 testing
- **WAVE:** 102 images, 72 training, and 30 testing



Implementation using CNN

To implement the prediction model we throw deep learning and Convolutional Neural Networks (CNNs) at the problem

So why ConvNet over other classification algorithms?

A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm that can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image, and be able to differentiate one from the other.
The pre-processing required in a ConvNet is much lower as compared to other classification algorithms.
Convolutional Neural Networks (ConvNets or CNNs) are a category of Neural Networks that have proven very effective in areas such as image recognition and classification.
Its built-in convolutional layer reduces the high dimensionality of images without losing its information

To start, we don't have much training data, only 72 images for training. When confronted with a lack of tracking data we typically apply data augmentation — but data augmentation in this context is also problematic.

Why Data Augmentation?

The dataset contains significantly fewer images per category in both train and test sets. Augmentation of the dataset artificially creates images for training and testing.

ImageDataGenerator is used to produce new images. The rotation range is set to 360 as the images are spiral and can be rotated any number of degrees without changing the image's meaning(horizontal flip & vertical flip).

While applying augmentation on certain transformations it significantly lowered the CNN model's lower accuracy. This is evidently seen while augmenting the wave data.

Now the training data's sample size has increased from 72 to 5112 images and hence ready to be trained.

Augmented data – Spiral drawing

Augmented data – Wave drawing

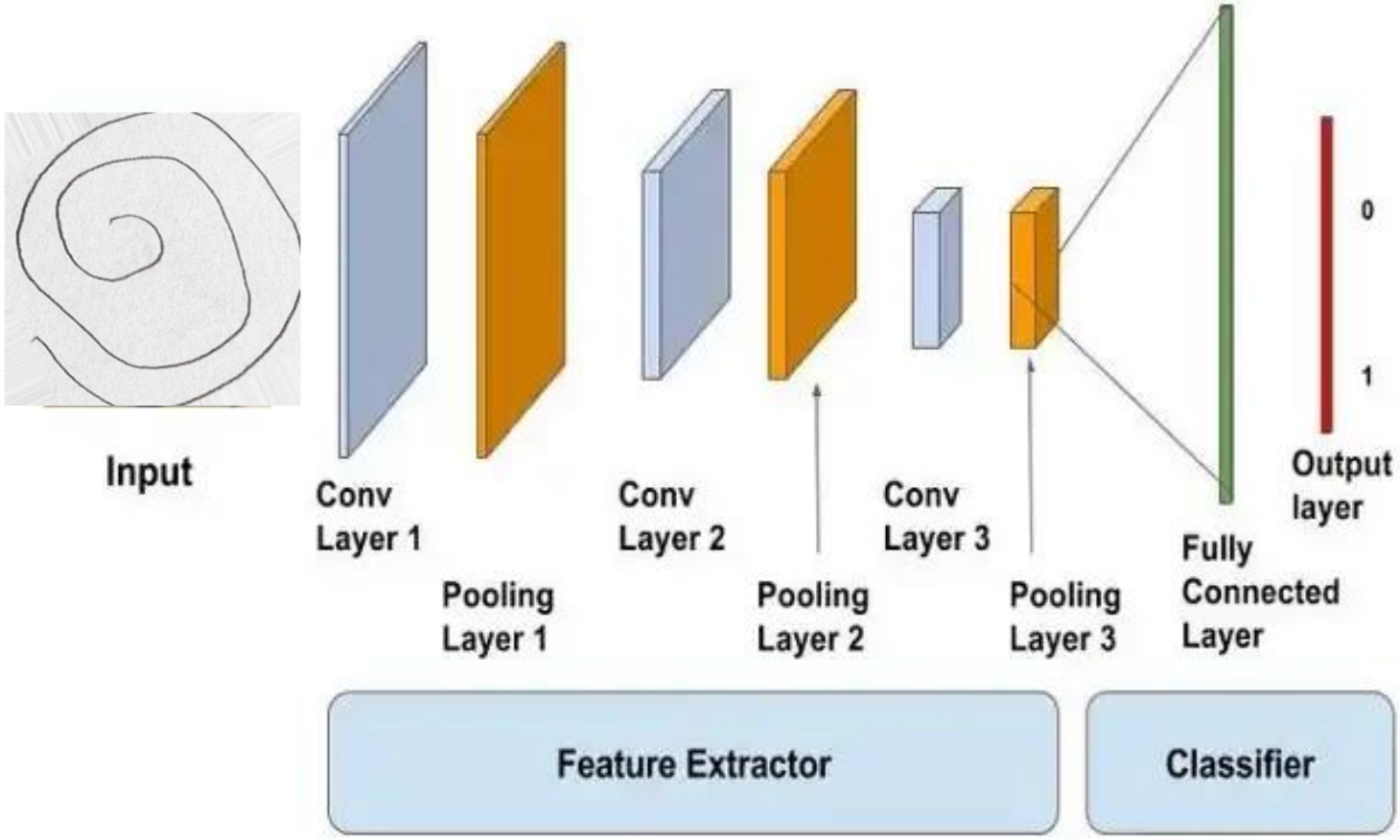
Wave Drawing by a Healthy Person

Wave Drawing by a Person having Parkinson's Disease

Training & Model fitting

The implementation uses a CNN model architecture with the following characteristics —
The Input layer in CNN contains image data. Image data is represented by three dimensional matrix. You need to reshape it into a single column.

•The model contains four Convolutional Layers with 128, 64, 32, and 32 filters, respectively.
•The convolutional layers contain filters with varying filter sizes.
•Also called feature extractor layer because features of the image are get extracted within this layer.
•Then we slide the filter over the next receptive field of the same input image by a Stride and do the same operation again. We will repeat the same process again and again until we go through the whole image. The output will be the input for the next layer.
•A MaxPool2D layer follows each convolutional layer.
•Max pooling is only way to reduce the spatial volume of input image.
•Two Fully Connected layers follow the convolutional block. Fully connected layer involves weights, biases, and neurons. It connects neurons in one layer to neurons in another layer. It is used to classify images between different category by training.
• ReLU activation is used to make all negative value to zero. Softmax is for multi-classification.(multi class classification)
•Output layer contains the label which is in the form of one-hot encoded.


Input
Conv Layer 1
Pooling Layer 1
Conv Layer 2
Pooling Layer 2
Conv Layer 3
Pooling Layer 3
Fully Connected Layer
0
1
Output layer
Feature Extractor
Classifier

Accuracy of Wave Model

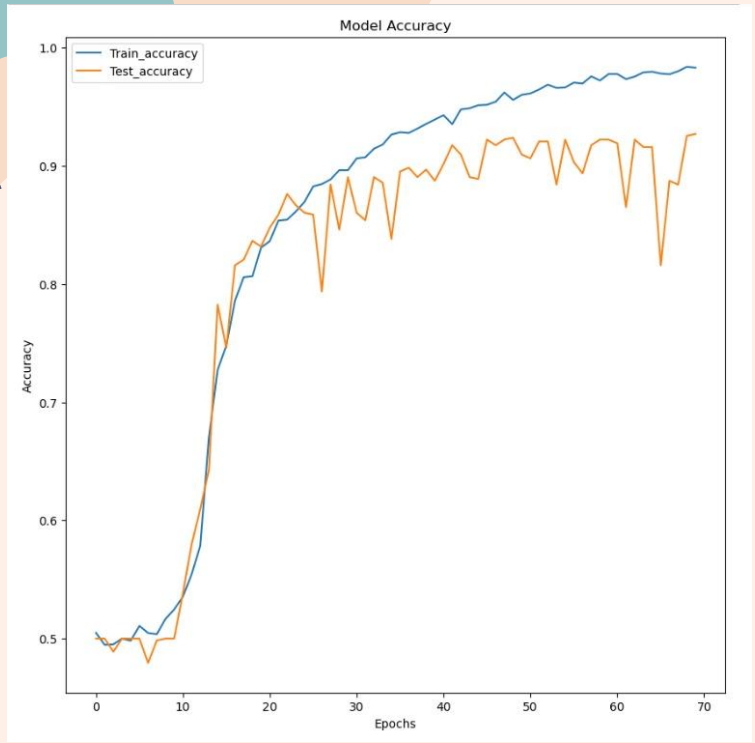
```
In [27]: hist = model.fit(x_train, y_train, batch_size=10, epochs=70, validation_data=(
```

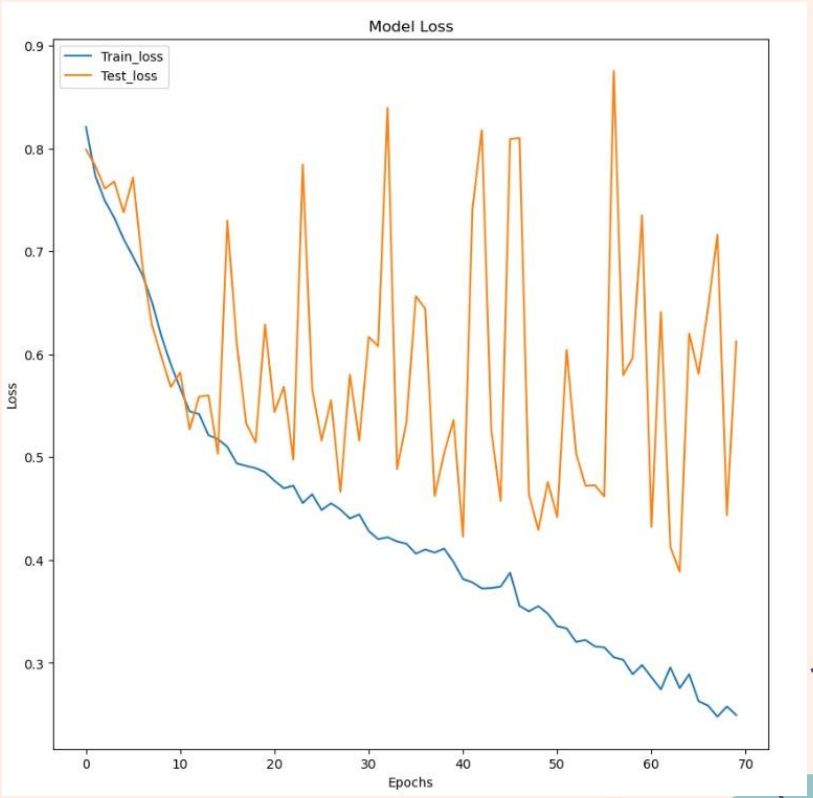
```
accuracy: 0.9249 - val_loss: 0.3885 - val_accuracy: 0.8683
Epoch 65/70
512/512 [==============================] - 59s 116ms/step - loss: 0.2890 -
accuracy: 0.9178 - val_loss: 0.6202 - val_accuracy: 0.8016
Epoch 66/70
512/512 [==============================] - 56s 109ms/step - loss: 0.2628 -
accuracy: 0.9294 - val_loss: 0.5809 - val_accuracy: 0.8190
Epoch 67/70
512/512 [==============================] - 57s 112ms/step - loss: 0.2585 -
accuracy: 0.9360 - val_loss: 0.6449 - val_accuracy: 0.8016
Epoch 68/70
512/512 [==============================] - 61s 119ms/step - loss: 0.2477 -
accuracy: 0.9353 - val_loss: 0.7161 - val_accuracy: 0.7937
Epoch 69/70
512/512 [==============================] - 57s 111ms/step - loss: 0.2577 -
accuracy: 0.9321 - val_loss: 0.4434 - val_accuracy: 0.8492
Epoch 70/70
512/512 [==============================] - 63s 123ms/step - loss: 0.2493 -
accuracy: 0.9335 - val_loss: 0.6125 - val_accuracy: 0.8127
```

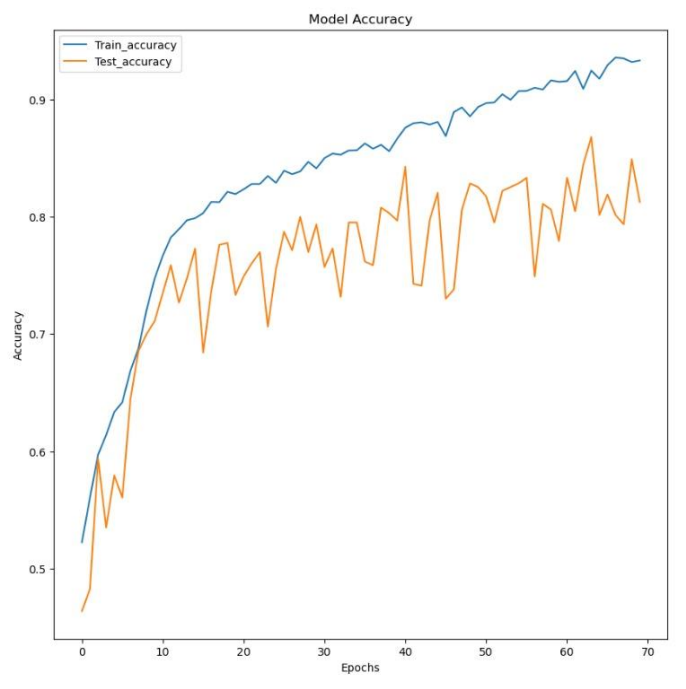
Accuracy of Spiral Model

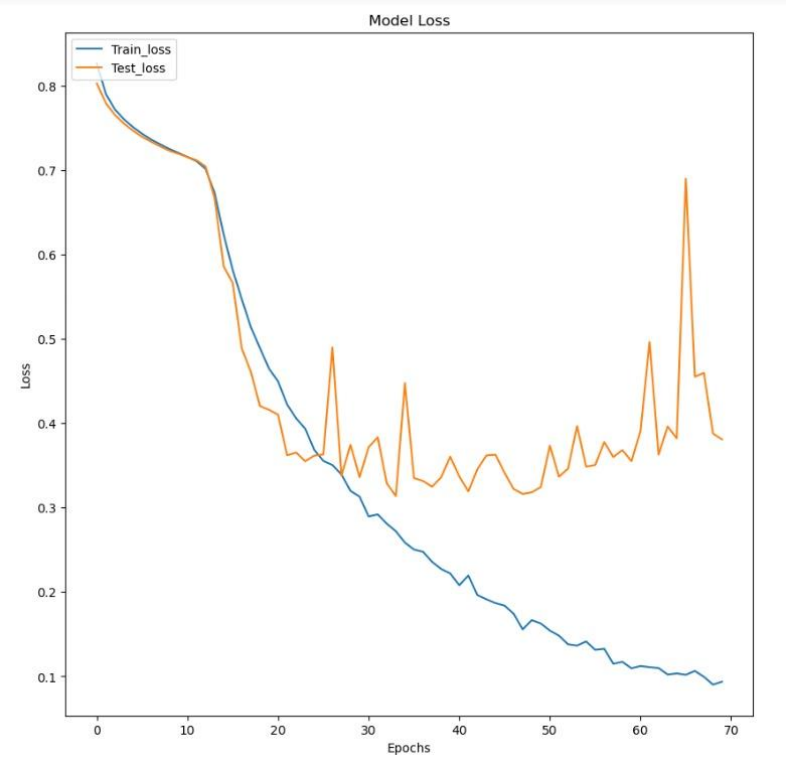
```
In [27]: hist = model.fit(x_train, y_train, batch_size=10, epochs=70, validation_data=(
```

```
ccuracy: 0.9791 - val_loss: 0.3963 - val_accuracy: 0.9159
Epoch 65/70
512/512 [==============================] - 46s 89ms/step - loss: 0.1035 - a
ccuracy: 0.9797 - val_loss: 0.3821 - val_accuracy: 0.9159
Epoch 66/70
512/512 [==============================] - 46s 90ms/step - loss: 0.1019 - a
ccuracy: 0.9781 - val_loss: 0.6900 - val_accuracy: 0.8159
Epoch 67/70
512/512 [==============================] - 48s 94ms/step - loss: 0.1064 - a
ccuracy: 0.9775 - val_loss: 0.4554 - val_accuracy: 0.8873
Epoch 68/70
512/512 [==============================] - 48s 93ms/step - loss: 0.0995 - a
ccuracy: 0.9800 - val_loss: 0.4598 - val_accuracy: 0.8841
Epoch 69/70
512/512 [==============================] - 48s 93ms/step - loss: 0.0902 - a
ccuracy: 0.9838 - val_loss: 0.3879 - val_accuracy: 0.9254
Epoch 70/70
512/512 [==============================] - 47s 92ms/step - loss: 0.0937 - a
ccuracy: 0.9830 - val_loss: 0.3809 - val_accuracy: 0.9270
```


Model Accuracy
Train_accuracy
Test_accuracy
Accuracy
Epochs
1.0
0.9
0.8
0.7
0.6
0.5
0
10
20
30
40
50
60
70


Model Loss
Train_loss
Test_loss
Loss
Epochs
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
10
20
30
40
50
60
70


Model Accuracy
Train_accuracy
Test_accuracy
Accuracy
Epochs
0.5
0.6
0.7
0.8
0.9
0
10
20
30
40
50
60
70


Model Loss
Train_loss
Test_loss
Loss
Epochs
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0
10
20
30
40
50
60
70

Resources

https://www.kaggle.com/code/brsdincer/parkinson-prediction-by-hand-drawing-dl-ml-process
https://www.kaggle.com/code/stpeteishii/parkinson-s-drawings-densenet201
https://medium.com/ai-techsystems/parkinsons-disease-detection-using-spiral-drawings-and-cnn-b82de9f3ed73
https://pyimagesearch.com/2019/04/29/detecting-parkinsons-disease-with-opencv-computer-vision-and-the-spiral-wave-test/
https://arxiv.org/ftp/arxiv/papers/2101/2101.05631.pdf
https://www.youtube.com/watch?v=eKy3KgRgDkQ
https://aircconline.com/csit/papers/vol12/csit121523.pdf
https://www.youtube.com/watch?v=eKy3KgRgDkQ
https://laptrinhx.com/detecting-parkinson-s-with-opencv-computer-vision-and-the-spiral-wave-test-3813156616/