EARLY DETECTION OF PARKINSON DISEASE USING MACHINE LEARNING

Submitted by,

Akhil Shaji

Erics Antony

1. INTRODUCTION

1.1. Overview

Parkinson's disease (PD) is a neurodegenerative movement disease where the symptoms gradually develop start with a slight tremor in one hand and a feeling of stiffness in the body and it became worse over time. It affects over 6 million people worldwide. At present there is no conclusive result for this disease by non-specialist clinicians, particularly in the early stage of the disease where identification of the symptoms are very difficult in its earlier stages. The proposed predictive analytics framework is a combination of K-means clustering and Decision Tree which is used to gain insights from patients. By using machine learning techniques, the problem can be solved with minimal error rate. Our proposed system provides accurate results by integrating spiral and wave drawing inputs of normal and Parkinson's affected patients. From these drawings Random forest classification algorithm is used which converts these drawings into pixels for classification and the extracted values are been matched with the trained database to extract various features and results are produced with maximum accuracy. Also OpenCV (Open Source Computer Vision Library) a library of programming functions mainly aimed at real-time computer vision was built to provide an infrastructure for computer vision applications and to accelerate the use of machine perception in the real time. Thus our output will showcase the early detection of the disease and can be able to increase the lifespan of the diseased patient with proper treatments and medications leads to peaceful life.

1.2. Purpose

The Parkinson's disease is due to a loss of neurons that produce a chemical messenger in the brain called dopamine. when there is a decrease in level of the amino acid named dopamine it leads to the abnormal brain activity, which leads to Parkinson's disease. The cause of Parkinson's disease is still a question mark, but several factors appear to play a role, including:

- Genes
- Environmental
- Triggers

As a result people suffer from this disease for many years before diagnosis. The estimated results have shown that there are 7-10 million people are affected by parkinson's disease worldwide. People with age above 50 are the one's who has the higher possibility of getting parkinson's disease but still an estimated 4 percentage of people who are under the age 50 are diagnosed with parkinson's disease. There is no cure or prevention for PD. However, the disease can be controlled in early stage. The data mining techniques is used as a effective way for early detection and diagnosis of the disease. Data mining techniques in medicine is a research area that combines sophisticated representational and computing techniques with the insights of expert physicians to produce tools for improving healthcare. Data mining is a statistical method for finding hidden patterns in datasets by constructing predictive or classification models that can be learned from past experience and applied in future cases, so there is a need for a more accurate, objective means of early detection, ideally one which can be used by individuals in their home setting.

2. LITERATURE SURVEY

2.1. Existing System

In existing system, PD is detected at the secondary stage only (Dopamine deficiency) which leads to medical challenges. Also doctor has to manually examine and suggest medical diagnosis in which the symptoms might vary from person to person so suggesting medicine is also a challenge. Thus the mental disorders are been poorly characterized and have many health complications. PD is generally diagnosed with the following clinical methods as,

- MRI or CT scan - Conventional MRI cannot detect early signs of Parkinson's disease
- PET scan - is used to assess activity and function of brain regions involved in movement
- SPECT scan - can reveal changes in brain chemistry, such as a decrease in dopamine

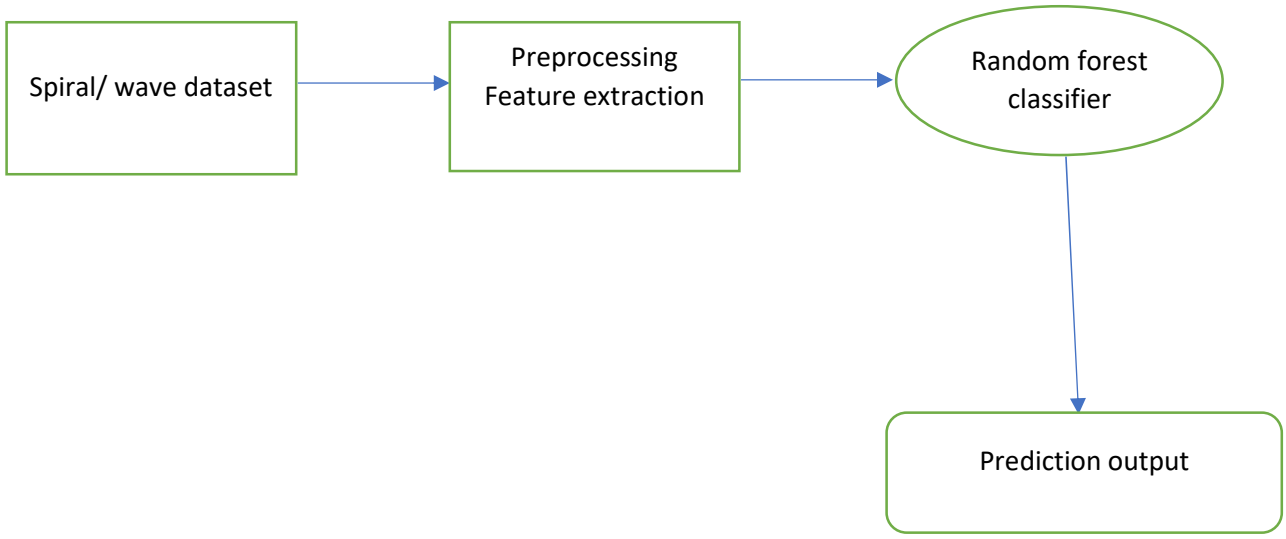
This results in a high misdiagnosis rate (up to 25% by non-specialists) and many years before diagnosis, people can have the disease. Thus existing system is not effective in early prediction and accurate medicinal diagnosis to the affected people.

2.2. Proposed System

By using machine learning techniques, the problem can be solved with minimal error rate. Also our proposed system provides accurate results by integrating spiral drawing inputs of normal and Parkinson's affected patients. We propose a hybrid and accurate results analyzing patient both wave and spiral drawing data's. Thus combining both the results, the doctor can conclude normality or abnormality and prescribe the medicine based on the affected stage.

3. THEORITICAL ANALYSIS

3.1. Block Diagram



3.2. Hardware / Software designing

Software Requirements:

- Software:
  - Spyder
  - Jupyter notebook
- Operating system: Windows 10
- Tools: Web browser
- Python Libraries: numpy, sklearn, pickle, cv2, flask. Imutils

Hardware Requirements:

- RAM: 4GB or above
- Storage: 30GB or above
- Processor: Any processor above 500MHz

4. EXPERIMENTAL INVESTIGATIONS

Spiral/wave drawing datasets of PD affected and unaffected patients collected by neurologists are obtained from Machine Learning repository. These are stored into the python environment as Testing and Training datasets and imported using necessary packages. Python is an open-source dynamic, high level, free and interpreted programming language. This supports object-oriented programming and procedural programming. Python is currently the most popular programming language for Machine Learning research and development. Spyder is an integrated development environment (IDE) primarily for the Python language, used in computer programming. Microsoft Visual Studio is a development environment by Microsoft. It is used to develop computer programs, websites, web applications, web services, and mobile apps.

1. Importing datasets into Spyder - Spiral drawing datasets of PD affected and unaffected patients collected by neurologists are obtained from Machine Learning repository. These are stored into the python environment as Testing and Training datasets and imported using necessary packages.

2. Pre-Processing – It involves image acquisition, pre-processing and segmentation. Preprocessing image is a way to improve image quality, so that the resulting image is better than the original one. The goal of image acquisition is to collect images having low noise when compared to HD images. The main advantage of this module is to have images with better clarity, low noise and distortion. The aim of segmentation is to make the representation of an image simpler or more easily analyzable.

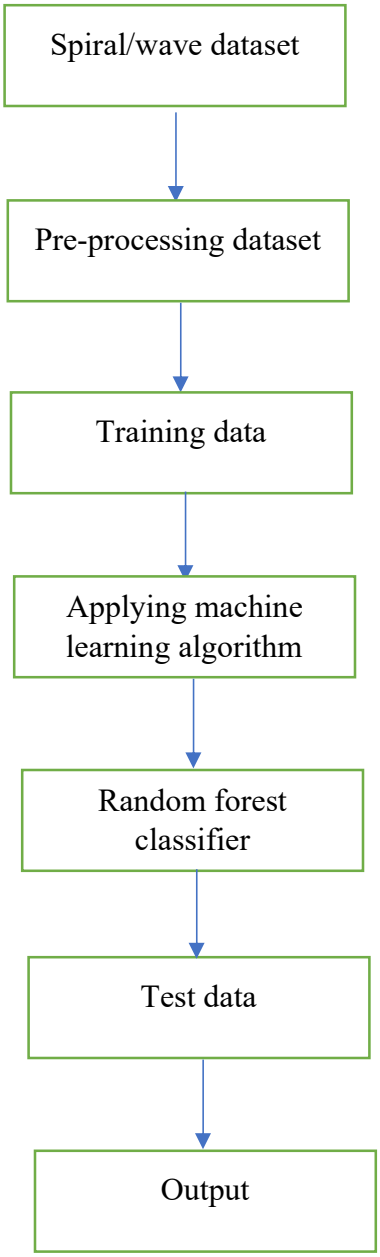
3. Feature extraction - In this project, mean filter and median filter are presented for processing of selecting the images. The median filter is a non-linear tool, while linear is the average filter. Mean filtering of smoothing images is fast, intuitive and easy to implement i.e. reduces the amount of variation in intensity between one pixel and the next. The median filter is normally used in a picture to reduce salt-and-pepper noise. It often does a better job than maintaining useful information in the picture than the mean filter. The median is determined by first sorting all the pixel values in numerical order from the surrounding area and then replacing the pixel that is considered with the middle pixel value. If there are even number of pixels in the neighborhood under consideration the sum of the two middle pixel values is used. Both mean and median filters are used to remove noise. This is used as the input for further analysis.

4. OpenCV library function - OpenCV (Open Source Computer Vision Library) was developed to provide an interface for computer vision applications and to facilitate the use of machine perception in the real time 5. Classification (Random Forest) – It is a supervised learning algorithm used for classification. Random forest algorithm builds decision trees on data samples, then obtains the prediction from each and finally sel ects the best solution by voting.It is an ensemble approach that is better than a single decision tree, as it eliminates o verfitting by averaging the outcome. Where we can find the confusion matrix with the help of confusion_matrix() function of sklearn, which is nothing but a table with two dimensions viz. “Actual” and “Predicted” and furthermore, both the dimensions have “True Positives (TP)”, “True Negatives (TN)”, “False Positives (FP)”, “False Negatives (FN)”,which calculates accuracy, specificity and sensitivity.

6. Predicted Output - Thus our hybrid architecture, integrating image processing (spiral drawing analyzing) using image processing technique, the predicted output based on Random forest Classification and confusion matrix is with an accuracy of 83%. Also it produces real-

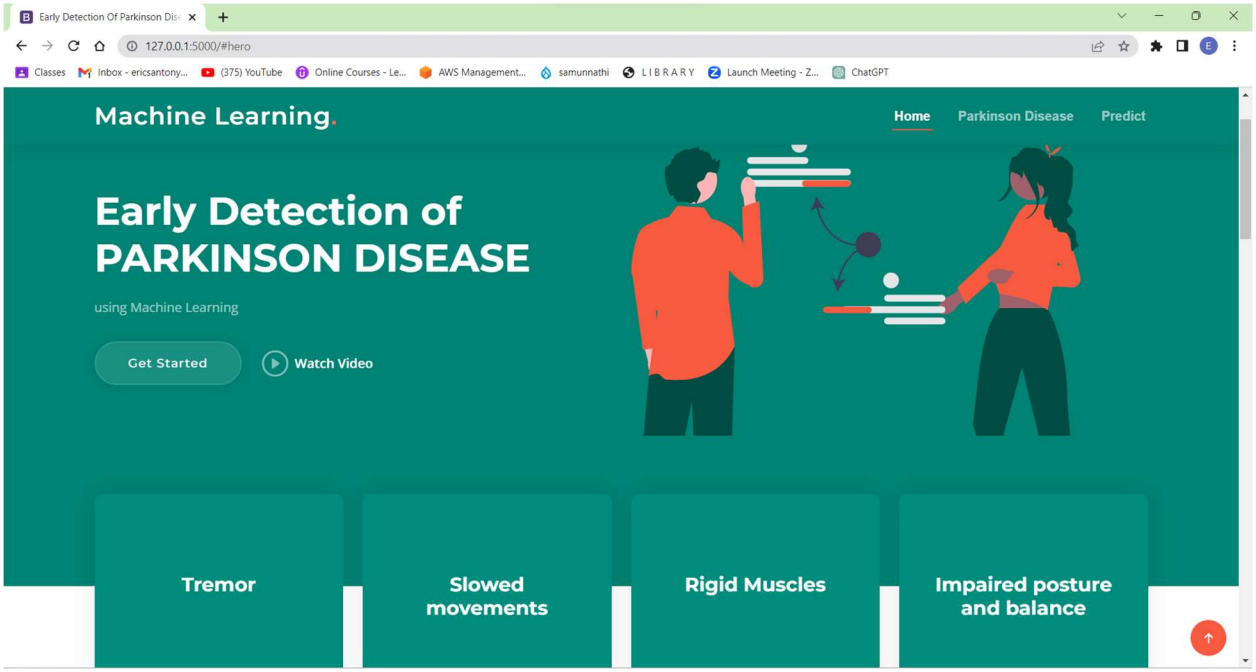
time accurate results by giving a person's spiral drawing as an input to the OpenCV function, that indicates whether a person is healthy or affected by Parkinson's.

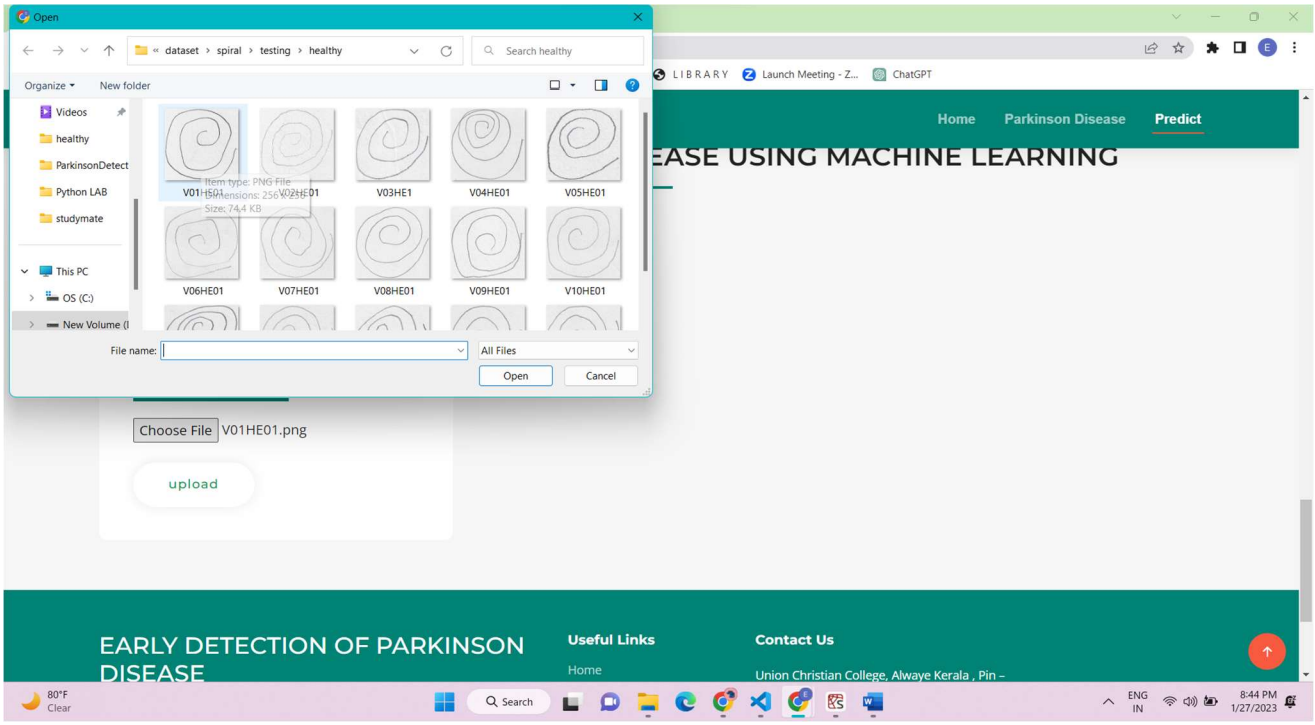
5. FLOWCHART

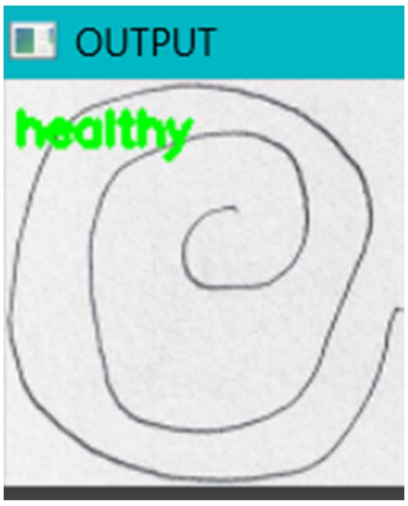


6. OUTPUT

The output of the project predicts whether the person is healthy or affected by parkinsons by analyzing the given input data which is spiral or wave data.







7. ADVANTAGES & DISADVANTAGES

Advantage: It's critical to correctly diagnose Parkinson's disease so that sufferers can receive the right treatment and counselling. Furthermore, recognizing Parkinson's disease early is critical since therapies like levodopa/carbidopa are more successful when given early in the disease. Non-pharmacological treatments, such as increased exercise, are also easier to implement in the early stages of Parkinson's disease and may help halt disease development.

Disadvantage: The Random Forest Classifier had the maximum accuracy of 83.12 percent. The diagnosis of bradykinesia and tremor, according to the data provided in section V, leads to tangible results for the early detection of this disease. Furthermore, it was discovered that the detection accuracy might be improved in two ways: by incorporating ensemble algorithms such as bagging, boosting, and voting, and by expanding the dataset size.

8. APPLICATION

Parkinson's disease (PD) is a progressive degenerative disease of the nervous system that affects movement control. This disease affects approximately 1% of the population over 60 years old, with a prevalence of approximately 250 per 100,000 persons, and an average age at onset of between 55 and 65. PD is a very complex disorder in which individual motor features

vary in their presence and severity over time. Early diagnosis of PD is essential, and so early diagnosis of PD is a subject of increasing research.

Previous studies have emphasized that the main challenge in the diagnosis of PD is the correct recognition of PD affected subjects in the early stages of the disease. Early diagnosis of PD can greatly affect the progression of the disease and the quality of life of the patient. Data mining has long been suggested as a potential tool for improving problems in early diagnosis and prediction, along with knowledge detection from medical repositories. Machine learning techniques have been effective in discovering hidden patterns in these data, and so expert systems developed by machine learning techniques can be used to assist physicians in the diagnosis and prediction of disease.

This Special Issue aims to collect recent developments in methods for Parkinson's disease diagnosis using machine learning. We seek both original research and review articles related to applications of machine learning for PD diagnosis and the considerable enhancements in the accuracy and cost-effectiveness of medical and health care services for PD such applications bring.

Other areas include;

- Used to detect Dementia at early stage.
- Used to detect neurodegenerative disorders.
- Used for clinical diagnosis for patients above 50 years.

9. CONCLUSION

Parkinson's disease is the second most dangerous neurodegenerative disease which has no cure till now and to make it reduce prediction is important. In this project, we have used prediction model to predict the Parkinson's disease which are Machine Learning Techniques i.e. Random Forest Classifier. The dataset is trained using this model and we also compared these different models built using different methods and identifies the best model that fits. The aim is to use various evaluation metrics such as Accuracy, Precision, Recall, Specificity, F1-score, LR+, LR- and Youden score that produce the predicts the disease efficiently. We have used the spiral and wave dataset that contains features of the patients. The dataset consists of adequate amount a data to train the model. The models are built using the five best features which were identified by feature selection. From this result, Random Forest Classifier with an accuracy of 83%. This system we designed can make the predictions of the Parkinson's disease.

10.FUTURE WORK

In future, these models can be trained with different datasets that have best features and can be predicted more accurately. If the accuracy rate increases, it can be used by the laboratories and hospitals so that it is easy to predict in early stages. This model can be also used with different medical and disease datasets. In future the work can be extended by building a hybrid model that can find more than one disease with an accurate dataset and that dataset has common features of two diseases. In future the work can extended to build a model that may extract more important features among all features in the dataset so that it produce more accuracy.

11.BIBLIOGRAPHY

- Smartinternz website
- W3 schools
- Youtube
- Stackoverflow

APPENDIX

```
# -*- coding: utf-8 -*-
"""
Created on Thu Jan 26 10:36:44 2023

@author: erics
"""

from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import confusion_matrix
from skimage import feature
from imutils import paths
import numpy as np
import cv2
import os
import pickle
import random
import matplotlib.pyplot as plt

def quantify_image(image):
    features = feature.hog(image, orientations=9,
                           pixels_per_cell=(10, 10), cells_per_block=(2, 2),
                           transform_sqrt=True, block_norm="L1")
    return features

def load_split(path):
    print(path)
    imagePaths = list(paths.list_images(path))
    print(imagePaths)
    data = []
```

```
    labels = []
    for imagePath in imagePaths:
        label = imagePath.split(os.path.sep)[-2]
        image = cv2.imread(imagePath)
        image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        image = cv2.resize(image, (200, 200))
        image = cv2.threshold(image, 0, 255,
                        cv2.THRESH_BINARY_INV | cv2.THRESH_OTSU)[1]
        features = quantify_image(image)
        data.append(features)
        labels.append(label)
    return (np.array(data), np.array(labels))


def train_model(dataset):
    print(dataset)
    path = "D:\studymate\MCA\sem 3\Extenship\ParkinsonDetection\dataset\\" + dataset
    trainingPath = os.path.sep.join([path, "training"])
    testingPath = os.path.sep.join([path, "testing"])
    print(trainingPath)
    (trainX, trainY) = load_split(trainingPath)
    (testX, testY) = load_split(testingPath)
    le = LabelEncoder()
    trainY = le.fit_transform(trainY)
    testY = le.transform(testY)
    model=RandomForestClassifier(n_estimators=100)
    model=model.fit(trainX, trainY)
    pickle.dump(model,open('parkPredict.pkl','wb'))
    predictions = model.predict(testX)
    cm = confusion_matrix(testY, predictions).ravel()
    tn, fp, fn, tp = cm
    accuracy = (tp + tn) / float(cm.sum())
```

```
    sensitivity= tp / float(tp + fn)
    specificity = tn / float(tn + fp)
    print(accuracy)
    print(sensitivity)
    print(specificity)
    return model

def test_prediction(model, testingPath):
    testingPaths = list(paths.list_images(testingPath))
    output_images = []
    for _ in range(25):
        image = cv2.imread(random.choice(testingPaths))
        output = image.copy()
        output = cv2.resize(output, (128, 128))
        image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        image = cv2.resize(image, (200, 200))
        image = cv2.threshold(image, 0, 255,
                              cv2.THRESH_BINARY_INV | cv2.THRESH_OTSU)[1]
        features = quantify_image(image)
        preds = model.predict([features])
        label = "Parkinsons" if preds[0] else "Healthy"
        color = (0, 255, 0) if label == "Healthy" else (0, 0, 255)
        cv2.putText(output, label, (3, 20), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
                    color, 2)
        output_images.append(output)
    plt.figure(figsize=(20, 20))
    for i in range(len(output_images)):
        plt.subplot(5, 5, i+1)
        plt.imshow(output_images[i])
        plt.axis("off")
    plt.show()
```

```
spiralModels = train_model('spiral')
testingPath = os.path.sep.join(["D:\studymate\MCA\sem
3\Extenship\ParkinsonDetection\dataset\spiral", "testing"])
print(testingPath)
    test_prediction(spiralModels, testingPath)
```