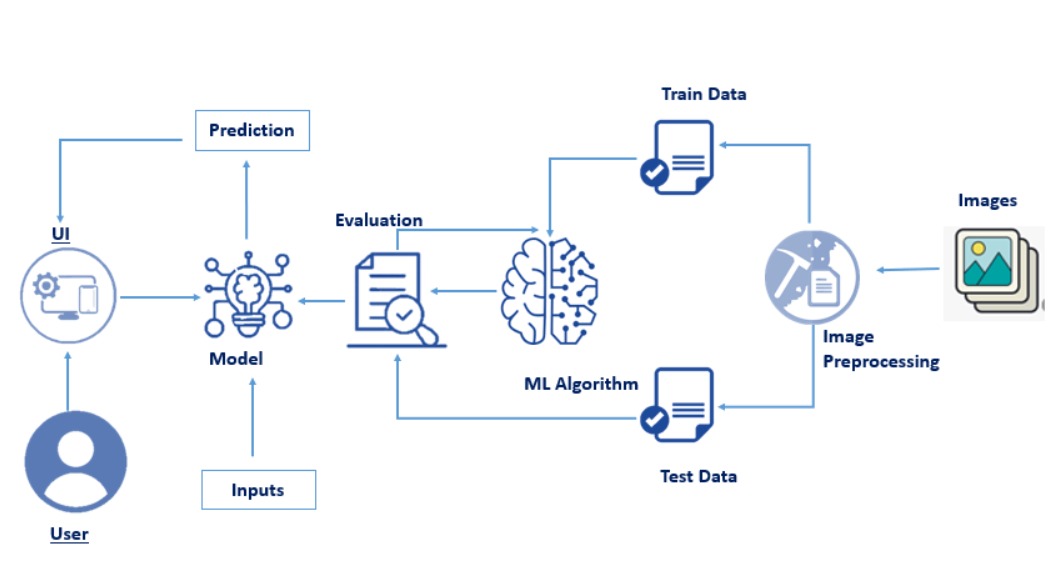
**Detecting Parkinson’s Disease Using Machine Learning**

More than 10 million people are living with Parkinson’s Disease worldwide, according to the Parkinson’s Foundation. While Parkinson’s cannot be cured, early detection along with proper medication can significantly improve symptoms and quality of life.

The researchers found that the drawing speed was slower and the pen pressure is lower among Parkinson’s patients. One of the indications of Parkinson’s is tremors and rigidity in the muscles, making it difficult to draw smooth spirals and waves. It is possible to detect Parkinson’s disease using the drawings alone instead of measuring the speed and pressure of the pen on paper. Our goal is to quantify the visual appearance(using HOG method) of these drawings and then train a machine learning model to classify them. In this project, We are using, Histogram of Oriented Gradients (HOG) image descriptor along with a Random Forest classifier to automatically detect Parkinson’s disease in hand-drawn images of spirals and waves.

**Technical Architecture:**



### Project Objectives

**By the end of this project:**

* You’ll be able to understand the problem to classify if it is a regression or a classification kind of problem.
* You will be able to know how to pre-process the image by using different data pre-processing techniques.
* you will be able to learn how to use OpenCV and machine learning to automatically detect Parkinson’s disease in hand-drawn images of spirals and waves
* You will be able to know how to find the accuracy of the model.
* You will be able to build web applications using the Flask framework.

### Project Flow

* User interacts with the UI (User Interface) to upload the image as input
* The uploaded image is analyzed by the model which is integrated
* Once the model analyzes the uploaded image, the prediction is showcased on the UI and OpenCV window

**To accomplish this, we have to complete all the activities and tasks listed below**

* Data Collection
  + Collect the dataset or Create the dataset
* Image Preprocessing.
* Importing the required libraries
* Loading Train data and Test data
* Quantifying images
* Label Encoding
* Model Building
  + Training the model
  + Testing the model
  + Model Evaluation
  + Saving the model
* Application Building
  + Create an HTML file
  + Build Python Code

### Pre-Requisites

#### In order to develop this project we need to install the following software/packages:

**Anaconda Navigator :**

Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning related applications. It can be installed on Windows, Linux, and macOS. Conda is an open-source, cross-platform, package management system. Anaconda comes with so very nice tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code. For this project, we will be using Jupyter notebook and Spyder

To install Anaconda navigator and to know how to use Jupyter Notebook & Spyder using Anaconda watch the video

<https://youtu.be/5mDYijMfSzs>

**To build Machine learning models you must require the following packages**

* **Numpy:**
* It is an open-source numerical Python library. It contains a multidimensional array and matrix data structures and can be used to perform mathematical operations
* **Scikit-learn:**
* It is a free machine learning library for Python. It features various algorithms like support vector machine, random forests, and k-nearest neighbours, and it also supports Python numerical and scientific libraries like NumPy and SciPy
* **Scikit-image**
* Scikit-image or skimage,  is an open-source Python package designed for image preprocessing.

* **Install imutils**
* Imutils are a series of convenience functions to make basic image processing functions such as translation, rotation, resizing, and displaying Matplotlib images easier with OpenCV
* Open anaconda prompt and type this command

**“pip install imutils”**

* **OpenCV**
* [OpenCV](https://en.wikipedia.org/wiki/OpenCV) is a library of programming functions mainly aimed at real-time computer vision. Here, OpenCV is used to capture frames by accessing the webcam in real-time.
* Open anaconda prompt and type this command

**“pip install opencv-contrib-python”**

* **Flask:**

Web framework used for building Web applications

If you are using anaconda navigator, follow below steps to download required packages:

* Open anaconda prompt.
* Type **“pip install numpy”**and click enter.
* Type **“pip install scikit-image”** and click enter.
* Type **“pip install imutils”** and click enter.
* Type**“pip install scikit-learn”**and click enter.
* Type **“pip install opencv-contrib-python”**and click enter.
* Type **“pip install Flask”** and click enter.

<https://youtu.be/akj3_wTploU>

If you are using Pycharm IDE, you can install the packages through the command prompt and follow the same syntax as above.

### Prior Knowledge

#### One should have knowledge of the following Concepts

Please refer to the videos below to gain required knowledge to complete the project.

<https://youtu.be/kE5QZ8G_78c>

<https://youtu.be/6za9_mh3uTE>

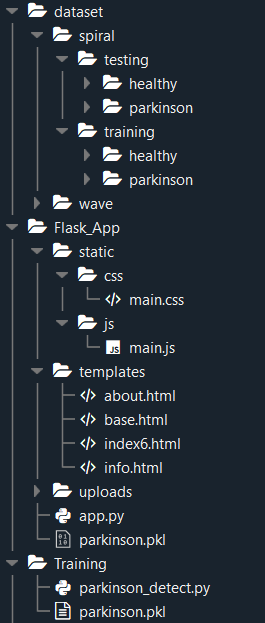
<https://youtu.be/nxFG5xdpDto>

<https://youtu.be/4ESLTAd3IOM>

<https://youtu.be/lj4I_CvBnt0>

### Project Structure

Let us introduce you to the main project folder downloaded by you in prerequisites.



* Training folder contains:
* Parkinson\_detect.py
* model file “parkinson.pkl”
* Flask App folder contains:
* static folder with the style sheets and the image required
* templates folder with the HTML pages
* app.py, a python script (Flask file) for server side computing.

### Data Collection

ML depends heavily on data, without data, it is impossible for a machine to learn. It is the most crucial aspect that makes algorithm training possible. In Machine Learning projects, we need a training data set. It is the actual data set used to train the model for performing various actions.

### Download The Dataset

You can collect datasets from different open sources like **kaggle.com, data.gov, UCI machine learning** repository etc.

The dataset is first broken down into spiral and wave. Each folder is further split into testing and training. Finally, our images reside in healthy or Parkinson folders.

The healthy folder consists of geometric images drawn by a healthy person.

The Parkinson folder consists of geometric images drawn by Parkinson person.

Please refer to the link given below to download the data set 



..

..

<https://drive.google.com/drive/folders/1Y3m9E98m33v6L9AqWWgHlzvMA_TsyQBi?usp=sharing>