A LIGHTWEIGHT SYSTEM TO DETECT PARKINSON'S DISEASE USING FACIAL MOTION ANALYSIS AND GRADIENT BOOSTING

Jiaheng Su [1] , Marisabel Chang [2]

[1] Viewpoint School, 23620 Mulholland Highway, Calabasas, California 91302
[2] California State Polytechnic University, Pomona, CA 91768

ABSTRACT

Parkinson's disease diagnosis traditionally relies on subjective clinical evaluation and expensive medical equipment, resulting in prolonged wait times, substantial costs, and misdiagnosis affecting nearly 20% of cases. Many machine learning approaches require data from medical-grade imaging systems such as MRI, limiting accessibility. This paper presents a lightweight screening system utilizing facial movement analysis from standard video recordings to provide objective, accessible PD detection. The methodology processes videos through MediaPipe to generate facial mesh representations, extracting landmarks that are transformed into Action Unit features including eye aspect ratio, mouth aspect ratio, angles, velocity, and acceleration. A supervised Gradient Boosting classifier processes these features to distinguish PD patients from healthy controls. Experimental evaluation demonstrates 86.7% classification accuracy, substantially outperforming unsupervised K-Means clustering (46.7%). The proposed multi-region, dynamics-aware approach offers practical preliminary screening suitable for resource-limited clinical settings where specialist access remains constrained.

KEYWORDS

Parkinson ' s Disease, Facial Action Units, FaceMesh, Computer Vision, EAR and MAR

1. INTRODUCTION

Current Parkinson's disease diagnostic methods typically require medical-grade imaging equipment or rely on subjective evaluation by clinical specialists, introducing potential for diagnostic errors [1]. This process imposes substantial burdens including extended wait times averaging 2.75 years and costs reaching thousands of dollars, creating barriers particularly for patients in underserved regions [2].

Clinical diagnosis accuracy across studies averages approximately 80.6%, indicating that nearly 20% of cases result in misdiagnosis [3]. Furthermore, clinicopathologic studies report interrater reliability (Cohen's kappa) of only ~0.54 at initial visits, demonstrating merely moderate agreement among clinicians rather than near-perfect consensus [4]. This substantial subjectivity indicates the need for more objective diagnostic approaches.

The diagnostic process frequently necessitates multiple revisits, compounding temporal and financial burdens while delaying appropriate treatment initiation [5]. Patients cannot receive proper therapeutic interventions during extended diagnostic evaluation periods, potentially allowing disease progression that earlier treatment might mitigate.

David C. Wyld et al. (Eds): NLTM, CCNET, DBDM, AIMLA, CICS – 2026
pp. 35-44, 2026. CS & IT - CSCP 2026 DOI: 10.5121/csit.2026.160604

Three research methodologies addressing hypomimia-based Parkinson's disease detection were analyzed. The first study focused on dynamic cheek surface modeling, achieving 80.6% accuracy but limiting analysis to a single facial region. The second approach analyzed smile-related Action Units using statistical summaries (mean and variance), achieving 79.8% accuracy but reducing temporal sensitivity through aggregation. The third methodology compared classical machine learning and deep learning models using FaceMesh landmarks, achieving 82.6% with CNN architectures but requiring larger datasets and greater computational resources.

Each methodology aimed to objectively quantify facial expressiveness but exhibited limitations in spatial coverage, temporal sensitivity, or computational practicality. The proposed methodology improves upon these works by analyzing multiple AU regions across the entire face and explicitly modeling temporal dynamics through velocity and acceleration features. By employing Gradient Boosting rather than deep neural networks, competitive accuracy (86.7%) is achieved while maintaining computational efficiency suitable for resource-constrained deployment.

The proposed method introduces a convenient and objective approach for parkinsonism screening utilizing commercial-grade smartphone cameras combined with machine learning classification [6]. The system generates facial mesh representations using MediaPipe and analyzes extracted features through trained classifiers, providing data-driven diagnostic support that is inherently objective.

Smartphone-based acquisition makes diagnosis accessible to most patients while providing clinicians with quantitative, reproducible information [7]. By analyzing subtle facial movements often overlooked during traditional examination, the application can detect early signs of parkinsonism with improved sensitivity. Compared to traditional methods requiring in-person evaluation and specialized equipment, this solution is non-invasive, cost-effective, and suitable for remote deployment [8].

This accessibility makes the proposed system particularly valuable in regions with limited neurologist availability. Additionally, the machine learning framework enables continuous improvement as additional training data becomes available, progressively enhancing diagnostic accuracy over deployment lifetime [9].

The experimental evaluation assessed classification performance of supervised versus unsupervised learning for Parkinson's disease detection using facial motion features. Specifically, the study tested whether Gradient Boosting (supervised) would outperform K-Means clustering (unsupervised) when trained on features extracted from MediaPipeFaceMesh, including EAR, MAR, angles, velocity, and acceleration metrics.

The dataset was constructed by segmenting videos into fixed-length clips, extracting facial landmarks, computing AU-related features, and assigning diagnostic labels. Following dataset partitioning into training and testing subsets, model performance was evaluated using accuracy and confusion matrices.

Results demonstrated that Gradient Boosting significantly outperformed K-Means, achieving 86.7% accuracy compared to 46.7%. This outcome confirms that supervised models leverage labeled data to learn discriminative patterns effectively, whereas unsupervised clustering lacks explicit class information necessary for identifying subtle PD-characteristic facial motion patterns. The experiment successfully validated supervised learning approaches for facial motion-based PD classification.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Video Processing and Segmentation

Fixed-length segmentation ensures consistency and reliability in feature extraction and model training by standardizing the temporal baseline across all samples. Processing variable-length recordings would introduce inconsistencies in velocity and acceleration calculations, since differing numbers of frames can significantly alter derivative values. For example, additional frames may produce meaningfully different motion gradients, potentially leading to false positives or false negatives in region-specific metrics. By segmenting videos into fixed-length intervals, the system generates comparable feature representations across samples, thereby supporting more stable and robust classifier training.

2.2. Facial Feature Extraction (FaceMesh/AU System)

The system utilizes MediaPipeFaceMesh to extract detailed facial landmarks from each video frame. Although MediaPipe does not directly compute Facial Action Units (AUs), anatomically relevant subsets of landmarks—corresponding to regions such as the eyes, mouth, eyebrows, and nose—are selected to approximate AU-related regions. Temporal dynamics are then derived from the trajectories of these landmarks by calculating metrics including Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), velocity, acceleration, and slope [18]. These features capture subtle motion patterns over time, enabling the identification of reduced facial expressivity characteristic of hypomimia.

2.3. Machine Learning Classification

Supervised learning methods such as Gradient Boosting outperform unsupervised approaches like K-Means because they utilize diagnostic labels to directly learn discriminative patterns aligned with the classification objective. In contrast, unsupervised clustering identifies inherent data groupings without reference to clinical categories, meaning the resulting clusters may not correspond to medically meaningful distinctions. Experimental results indicate that facial motion differences between Parkinson's disease patients and healthy subjects are subtle and distributed across multiple features, requiring explicit supervision to be effectively captured. By leveraging labeled data, supervised models can optimize decision boundaries specifically for diagnostic separation, leading to superior performance.

3. SOLUTION

This section describes the system architecture, technologies employed, and operational workflow of the proposed Parkinson's disease detection application.

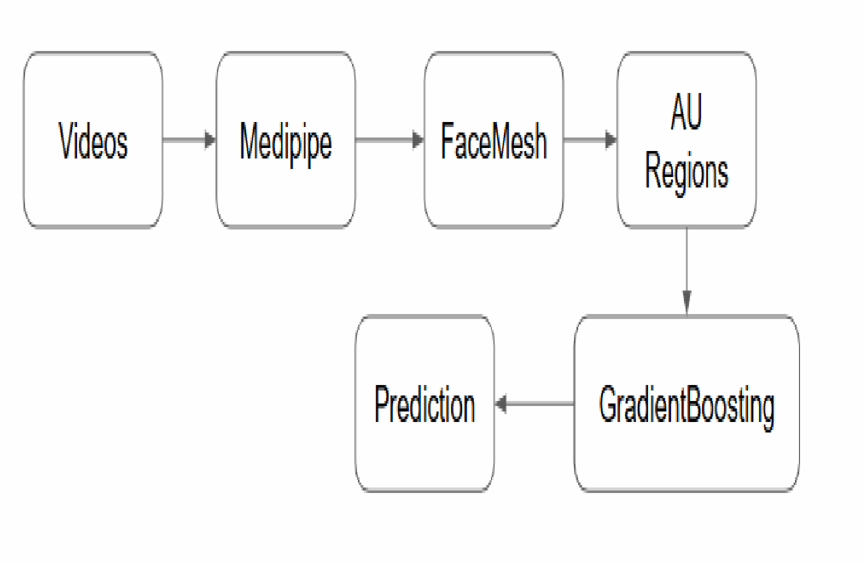

Figure 1. System architecture flowchart depicting the complete processing pipeline from video input to classification output

The proposed system processes video input through a sequential pipeline designed for efficient facial motion analysis. Initially, video recordings are converted to facial mesh representations using MediaPipe'sFaceMesh module. Subsequently, Action Unit (AU) regions are extracted from the mesh data and organized into structured DataFrames for analysis.

The feature extraction phase computes multiple metrics from the landmark data, including velocity, acceleration, and angular measurements that characterize facial motion dynamics. These calculated features serve as input to a Gradient Boosting machine learning classifier, which performs binary classification to distinguish between Parkinson's disease patients and healthy control subjects [16].

The following subsections detail three primary system components: video processing, feature extraction, and machine learning classification.

Video data is sourced from publicly available YouTube recordings of individuals with confirmed Parkinson's disease diagnoses alongside healthy control subjects. The preprocessing pipeline converts raw video into facial mesh representations and segments the data into discrete frames. Each video segment receives appropriate labeling indicating the subject's health status (Parkinson's or healthy) encoded as boolean values for subsequent supervised learning [17].

```
import cv2
import mediapipe as mp
import pandas as pd
import numpy as np
from pathlib import Path
import os
os.makedirs("facemeshes",exist_ok=True)

# Take your test video and paste it here.
def video_to_facemesh(video_path,hea):
   vid_capture = cv2.VideoCapture(video_path)
   mp_face_mesh = mp.solutions.face_mesh
   face_mesh = mp_face_mesh.FaceMesh()

   xx=[]
   index =0
   five_sec = 30 * 5
   while(vid_capture.isOpened()):


       ret, frame = vid_capture.read()
       if ret == False:
           print(index)
           break
       height, width=frame.shape[:2]
       border_size = 50

       framed_img = np.zeros((height, width, 3), dtype=np.uint8)
       results = face_mesh.process(frame)
       if results.multi_face_landmarks:
           xxxxx={}
           for face_landmarks in results.multi_face_landmarks:
               for i, landmark in enumerate(face_landmarks.landmark):
                   x, y, z = int(landmark.x * frame.shape[1]), int(landmark.y * frame.shape[0]),
int(landmark.z * frame.shape[1])
                   cv2.circle(framed_img, (x, y), 2, (0, 255, 0), -1)
                   xxxxx[f"x{i}"]=x
                   xxxxx[f'y{i}']=y
                   xxxxx[f'z{i}']=z
           xx.append(xxxxx)

       #cv2.imshow("frame",framed_img)

       #if cv2.waitKey(1) & 0xFF == ord('q'):
          # break


       if index != 0 and index % five_sec == 0:
           df=pd.DataFrame(xx)
           if hea:
               healthy=Path(video_path).stem+f"{index//five_sec}_h_"+".csv"
           else:
               healthy=Path(video_path).stem+f"{index//five_sec}"+"_pd_"+".csv"
           df.to_csv("facemeshes/"+healthy)
           xx=[]
       index+=1


   # Release the objects
   vid_capture.release()



if __name__ == '__main__':

   vid_path = 'x.mp4'
   video_to_facemesh(video_path=vid_path,hea=True)
```

Figure 2. Video segmentation and facial landmark extraction algorithm pseudocode

The video processing algorithm performs temporal segmentation and facial landmark extraction. The code initially separates input video into individual frames, subsequently organizing them into 5-second segments corresponding to approximately 150 frames based on the standard 30 fps frame rate. For each frame, MediaPipe generates a three-dimensional facial mesh from which x, y, and z coordinates are extracted for all 468 landmarks [19].

The algorithm maintains a running collection of landmark data that is periodically exported when segment boundaries are reached. Each exported CSV file receives a filename encoding both the source video identifier and subject health status label, facilitating subsequent supervised learning. Upon processing completion, video capture resources are released to prevent memory leaks.

The dataset construction component extracts clinically relevant features from facial landmarks corresponding to specific Action Unit regions [20]. The primary AU regions analyzed include the mouth, eyebrows, and eyes, which are known to exhibit reduced expressiveness in Parkinson's disease patients.

Following landmark extraction, the system computes Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), jaw angles, eyebrow movement metrics, and slope measurements for both left and right facial regions. These static features are augmented with temporal derivatives including velocity and acceleration to capture motion dynamics.

```
mouth=[61,0,17,291,13,14]
righteye=[33,159,145,133,158,153]
lefteye=[386,374,362,263, 385, 380]
Lefteyebrow=[ 285,282,276]
Righteyebrow=[ 225,223,22]
Chin=[152]
nose=[1]


import numpy as np
import pandas as pd
from pathlib import Path
import os
import glob
os.makedirs("dataset",exist_ok=True)
def read_data_Set(csv_path):
   df=pd.read_csv(csv_path)
   result=pd.DataFrame()
   result["EARL"]=df.apply(lambda row:EAR(lefteye,row),axis=1)
   result["EARR"]=df.apply(lambda row:EAR(righteye,row),axis=1)
   result["mar"]=df.apply(lambda row:mar(mouth,row),axis=1)
   result["jaw_angle"]=df.apply(lambda row:jaw_angle(row,Chin,mouth),axis=1)
   result["eyebrow_movementL"]=df.apply(lambda
row:eyebrow_movement(row,lefteye,Lefteyebrow),axis=1)
   result["eyebrow_movementR"]=df.apply(lambda
row:eyebrow_movement(row,righteye,Righteyebrow),axis=1)
   result["slopeL"]=df.apply(lambda row:slope(row,Righteyebrow),axis=1)
   result["slopeR"]=df.apply(lambda row:slope(row,Lefteyebrow),axis=1)
   labels=result.columns.to_list()
   vel= pd.DataFrame()
   accel=pd.DataFrame()
   for i in labels:
       if "slope" not in i:
           vel[f'{i}_vel'] = velocity(result[i],1/30)
           accel[f'{i}accel']=acceleration(vel[f'{i}_vel'],1/30)
   result=pd.concat([result, vel, accel],axis=1)
   outputpath=Path(csv_path).stem+".csv"
   result.to_csv("dataset/"+outputpath)
```

Figure 3.Feature extraction pipeline for AU-based temporal metrics computation

The feature extraction algorithm constructs a comprehensive dataset by computing multiple metrics from facial mesh landmarks. The code first defines arrays of MediaPipe landmark indices corresponding to key facial regions including the mouth, bilateral eyes, eyebrows, chin, and nose reference points.

For each input CSV file containing frame-by-frame landmark coordinates, the algorithm computes regional features: Eye Aspect Ratio (EAR) quantifies eye openness, Mouth Aspect Ratio (MAR) measures oral aperture, jaw angle captures mandibular position, and eyebrow metrics track vertical displacement and inclination [15]. Temporal dynamics are computed using a fixed frame rate of 30 fps, calculating both velocity (first derivative) and acceleration (second derivative) to detect movement speed abnormalities characteristic of hypomimia.

All computed features are concatenated into a unified DataFrame and exported for machine learning classification.

The classification component employs Gradient Boosting, implemented via the scikit-learn library, to perform supervised binary classification [14]. The model is trained on the feature dataset constructed from processed video segments, learning to distinguish patterns characteristic of Parkinson’s disease versus healthy facial motion.

```
def flat(path):
    df=pd.read_csv(path)
    filename=Path(path).stem
    v = df.unstack().to_frame().sort_index(level=1).T
    v.columns = [f"{col}_{i}" for col, i in v.columns]
    v["videopath"]=path
    #print(filename)
    if "_h_" in filename:
        v["target"]=0#healthy
    else:
        v["target"]=1#pd

    return v.reset_index(drop=True)
def get_training_dataset():
    df=pd.DataFrame()
    for path in glob.glob("dataset/*"):
        df1=flat(path)
        df=pd.concat([df,df1],ignore_index=True)
    return df
```

```
from sklearn.cluster import KMeans
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier

from anglefunction import get_training_dataset
from sklearn.metrics import f1_score
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
import matplotlib.pyplot as plt
df=get_training_dataset()
print(df.shape)
df = df.loc[:, ~df.columns.str.contains("^Unnamed")]
#print(df.shape)
df = df.fillna(0)
#print(df.isnull().sum())
y=df["target"]
X=df.drop(columns=["target","videopath"])
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33)
clf = GradientBoostingClassifier(n_estimators=100, learning_rate=1,max_depth=1, random_state=0).fit(X_train, y_train)
score=clf.score(X_test, y_test)
print(score)
```

Figure 4.Gradient Boosting classification pipeline and dataset preprocessing workflow

The classification pipeline begins by importing necessary scikit-learn modules for model training and evaluation. The get_training_dataset function from the anglefunction module aggregates all processed CSV files, reshaping each video segment's temporal data by unstacking rows into a single wide feature vector where each original feature is indexed by frame number.

Label encoding assigns binary values: 0 represents healthy subjects while 1 indicates Parkinson's disease. Following dataset aggregation, preprocessing removes unnamed index columns and fills missing values with zeros [13]. The dataset is partitioned with 67% allocated for training and 33% reserved for evaluation.

The Gradient Boosting classifier is instantiated with optimized hyperparameters: 100 estimators provide ensemble complexity, learning rate of 1.0 enables aggressive parameter adjustment, maximum depth of 1 creates simple decision stumps, and random state ensures reproducibility. Model accuracy is computed on the held-out test set.

4. EXPERIMENT

4.1. Experiment 1

This experiment evaluates the classification performance of supervised versus unsupervised learning approaches for Parkinson's disease detection using facial motion features. Approximately 45 video segments were analyzed, with extracted features including facial mesh coordinates, acceleration, velocity, and AU region metrics. The comparative study examines Gradient Boosting (supervised) against K-Means clustering (unsupervised) to determine whether explicit diagnostic labels are necessary for effective classification.

To evaluate the effectiveness of supervised and unsupervised learning approaches for Parkinson's Disease classification, a controlled comparison was conducted. The Gradient Boosting classifier (supervised) was trained on labeled examples using a 67/33 train-test split, while K-Means clustering (unsupervised) was applied to the same feature space without access to diagnostic labels.

Both models received identical preprocessed feature matrices containing EAR, MAR, angular, velocity, and acceleration features. Performance was assessed using accuracy, F1 score, and confusion matrix analysis. This experimental design isolates the impact of learning paradigm while controlling for feature representation.

Model	Accuracy	F1 Score
Gradient Boosting	86.7%	0.865
K-Means Clustering	46.7%	0.297

Table 1. Classification performance comparison between supervised and unsupervised approaches

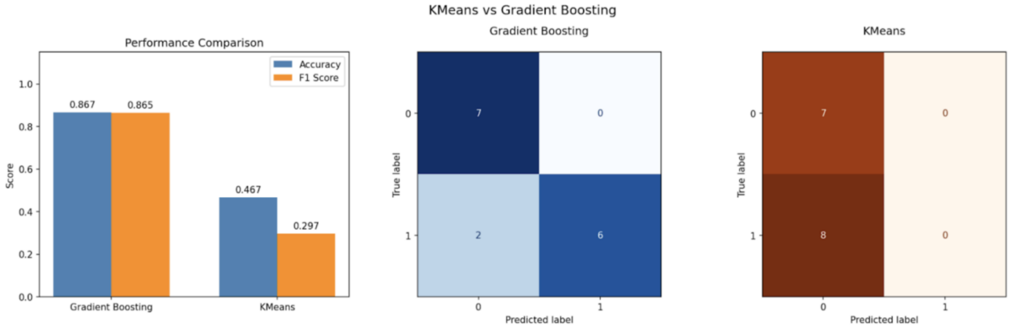

Figure 5. Comparison of Gradient Boosting versus K-Means classification performance showing accuracy and F1 scores with corresponding confusion matrices

The Gradient Boosting model demonstrated strong classification performance, achieving 86.7% accuracy with an F1 score of 0.865. Confusion matrix analysis reveals that the model correctly classified all healthy samples with zero false positives, while successfully identifying the majority of Parkinson's disease cases. A small number of PD samples were misclassified as healthy, representing false negatives that warrant attention in screening applications.

This performance indicates that the Gradient Boosting model effectively learns discriminative patterns from labeled training data, particularly excelling at identifying healthy subjects while maintaining strong sensitivity to PD-characteristic motion patterns. The model's ability to utilize labeled features enables construction of complex decision boundaries appropriate for distinguishing subtle facial motion differences.

In contrast, K-Means clustering achieved only 46.7% accuracy, performing near chance level. The unsupervised approach failed to identify meaningful clusters corresponding to diagnostic categories, confirming that explicit class labels are essential for effective PD detection using facial motion features.

5. RELATED WORK

Rios-Urrego et al. investigated dynamic cheek surface modeling for enhanced hypomimia detection in Parkinson's disease [10]. Their study analyzed cheek surface variability within video frames to evaluate disease presence, collecting data from 112 participants (32 PD patients, 80 healthy controls). Their machine learning approach achieved 80.6% prediction accuracy.

In contrast to their single-region focus on cheek dynamics, the proposed methodology analyzes multiple facial regions including eyes, mouth, and eyebrows, potentially capturing a broader range of hypomimia manifestations. Additionally, explicit temporal dynamics (velocity, acceleration) are computed rather than relying solely on spatial variability measures.

Adnan et al. proposed an AI-enabled screening framework that predicts Parkinson's disease status by analyzing smile expressions [11]. Participants were recorded during smile production, and Action Unit regions were extracted to achieve 79.8% classification accuracy. Their approach relies on statistical summaries including variance and mean values.

The proposed methodology differs by processing individual frames rather than aggregated statistics, explicitly computing velocity and acceleration to preserve temporal dynamics that statistical aggregation obscures. Furthermore, the proposed system does not require specific elicited expressions, enabling analysis of spontaneous facial behavior.

Castillo-Chica et al. compared classical machine learning and deep learning approaches for hypomimia-based PD classification using facial videos [12]. Their methodology employed FaceMesh landmark extraction with Support Vector Machine classification, while deep learning experiments using CNNs achieved 82.6% Unweighted Average Recall.

The proposed Gradient Boosting approach achieves competitive accuracy (86.7%) while maintaining substantially lower computational requirements than deep neural networks. Although deep learning models can capture complex hierarchical patterns, they require larger training datasets and greater computational resources, making Gradient Boosting more practical for deployment in resource-constrained clinical settings.

6. CONCLUSIONS

Several limitations warrant acknowledgment regarding facemesh quality and detection accuracy. MediaPipe generates facial landmarks through machine learning inference on pixel data, potentially missing extremely subtle movements that specialized depth sensors or radar-based systems could capture. Future implementations should evaluate alternative sensing modalities to obtain true motion measurements rather than machine-estimated approximations.

The current dataset comprises approximately 45 video segments, limiting statistical power for generalization assessment. Expanding training data through clinical partnerships would strengthen model robustness and enable evaluation across diverse demographic groups. Additionally, more sophisticated machine learning architectures including deep neural networks could be explored given sufficient training data, potentially improving classification sensitivity for challenging cases.

Future development priorities include mobile application deployment for practical clinical use, integration with electronic health record systems, and longitudinal studies evaluating the system's potential for monitoring disease progression and treatment response.

REFERENCES

[1] Postuma, Ronald B., et al. "MDS clinical diagnostic criteria for Parkinson's disease." Movement disorders 30.12 (2015): 1591-1601.

[2] Willis, Allison W., et al. "Neurologist care in Parkinson disease: a utilization, outcomes, and survival study." Neurology 77.9 (2011): 851-857.

[3] Rizzo, Giovanni, et al. "Accuracy of clinical diagnosis of Parkinson disease: a systematic review and meta-analysis." Neurology 86.6 (2016): 566-576.

[4] Rajput, D. R. "Accuracy of clinical diagnosis of idiopathic Parkinson's disease." Journal of neurology, neurosurgery, and psychiatry 56.8 (1993): 938.

[5] Espay, Alberto J., et al. "Technology in Parkinson's disease: challenges and opportunities." Movement Disorders 31.9 (2016): 1272-1282.

[6] Zhan, Andong, et al. "Using smartphones and machine learning to quantify Parkinson disease severity: the mobile Parkinson disease score." JAMA neurology 75.7 (2018): 876-880.

[7] Arora, Siddharth, et al. "Detecting and monitoring the symptoms of Parkinson's disease using smartphones: A pilot study." Parkinsonism & related disorders 21.6 (2015): 650-653.

[8] Bot, Brian M., et al. "The mPower study, Parkinson disease mobile data collected using ResearchKit." Scientific data 3.1 (2016): 1-9.

[9] Goetz, Christopher G., et al. "Movement Disorder Society‐sponsored revision of the Unified Parkinson's Disease Rating Scale (MDS‐UPDRS): scale presentation and clinimetric testing results." Movement disorders: official journal of the Movement Disorder Society 23.15 (2008): 2129-2170.

[10] Rios-Urrego, Cristian David, et al. "Dynamic cheek surface modeling for enhanced hypomimia detection in Parkinson's disease." Computers in Biology and Medicine 197 (2025): 110896.

[11] Adnan, Tariq, et al. "Unmasking Parkinson's Disease with Smile: An AI-enabled Screening Framework." arXiv preprint arXiv:2308.02588 (2023).

[12] Oliveira, Guilherme, et al. "Facial Expression Analysis in Parkinsons's Disease Using Machine Learning: A Review." ACM Computing Surveys 57.8 (2025): 1-25.

[13] Lugaresi, Camillo, et al. "Mediapipe: A framework for building perception pipelines." arXiv preprint arXiv:1906.08172 (2019).

[14] Friedman, Jerome H. "Greedy function approximation: a gradient boosting machine." Annals of statistics (2001): 1189-1232.

[15] Bologna, Matteo, et al. "Facial bradykinesia." Journal of Neurology, Neurosurgery & Psychiatry 84.6 (2013): 681-685.

[16] Bandini, Andrea, et al. "Analysis of facial expressions in parkinson's disease through video-based automatic methods." Journal of neuroscience methods 281 (2017): 7-20.

[17] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." the Journal of machine Learning research 12 (2011): 2825-2830.

[18] Dorsey, E. Ray, et al. "The emerging evidence of the Parkinson pandemic." Journal of Parkinson's disease 8.s1 (2018): S3-S8.

[19] Tsanas, Athanasios, et al. "Accurate telemonitoring of Parkinson's disease progression by non-invasive speech tests." Nature Precedings (2009): 1-1.

[20] Baltrušaitis, Tadas, Peter Robinson, and Louis-Philippe Morency. "Openface: an open source facial behavior analysis toolkit." 2016 IEEE winter conference on applications of computer vision (WACV). IEEE, 2016.