Deep Learning Approaches for Parkinson's Disease Detection

ARAVA JAYA LAKSHMI1, MAMIDALA SAI KUMAR2

#1Assistant Professor, Department of CSE, PBR Visvodaya Institute of Technology and Science, Kavali

#2 Assistant Professor, Department of CSE, PBR Visvodaya Institute of Technology and Science, Kavali

ABSTRACT:

Parkinson's disease is a neurological condition affecting millions of individuals worldwide. Parkinson's disease (PD) affects 60 percent of those over 50. Parkinson's disease sufferers have difficulty speaking and moving, making it difficult for them to travel for treatment and monitoring appointments. Early detection of Parkinson's disease facilitates therapy, allowing people to lead normal lives. The world's aging population emphasizes the importance of identifying Parkinson's disease early, remotely, and correctly. In recent years, machine learning approaches have demonstrated significant promise in the early identification and diagnosis of Parkinson's disease. In this project, we proposed a unique approach for the identification of Parkinson's illness utilizing machine learning techniques and the Xception architecture. . We specifically focus on the detection of Parkinson's disease using spiral and wave drawings, which are routinely employed in clinical practice as part of the diagnosis process. We compiled a dataset of spiral and wave drawings from people with and without Parkinson's disease. We preprocessed the data and trained our machine learning models using the Xception framework. Our models performed admirably, with a training accuracy of 95.34% and a validation accuracy of 93.000% for detecting Parkinson's disease from spiral drawings, and a training accuracy of 93.34% and a validation accuracy of 86.00% for detecting Parkinson's disease from wave drawings. Our findings show that machine learning and the Xception architecture have the potential to help detect and diagnose Parkinson's disease early on.

1.INTRODUCTION

Parkinson's Disease is a neurodegenerative disorder that affects millions of people worldwide. It is a progressive disease that affects movement, causing tremors, stiffness, and difficulty with balance and coordination. Early detection of Parkinson's Disease is crucial for effective treatment and management of the disease. However, current diagnostic techniques can be expensive, time-consuming, and often require specialized equipment and expertise.

In recent years, deep learning techniques have shown promising results in medical image analysis, including the detection of Parkinson's Disease. In this project, we propose a system for the early detection of Parkinson's Disease using Xception architecture. Specifically, we aim to detect Parkinson's Disease from spiral and wave drawings, which are commonly used in clinical settings as part of the diagnostic process.

The proposed system leverages the state-of-the-art Xception architecture, which has been shown to achieve high accuracy on a variety of image classification tasks. By using this architecture, we aim to develop a reliable and accurate system for the early detection of Parkinson's Disease, which could lead to better patient outcomes and quality of life.

Overall, this project aims to contribute to the ongoing efforts to improve the diagnosis and treatment of Parkinson's Disease, and to demonstrate the potential of deep learning and Xception architecture in medical image analysis..

2.LITERATURE REVIEW

Background: Motor complaints are the primary basis for a diagnosis of Parkinson's disease (PD), with costly and sometimes unavailable imaging methods like single photon emission computed tomography (SPECT) and M-iodobenzyl-guanidine heart scintiscan (MIBG) providing additional support. In this study, we looked at the research on electroencephalography (EEG) evaluations for quiescent state and muscle activation to identify PD with ML methods. Methods: The study was conducted according to the standards established by the Recommended Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA). Detailed notes were taken on the key features and outcomes of all releases up until May 2020. The findings included nine research. Seven participants used EEG during repose, and two participated in EEG during movement. The majority of research (83.3%) relied on subsymbolic models. Accuracy ranged from 0.62 to 99.62% when classifying PD. There was a significant deal of variation in the EEG cleansing procedure and the derived features. On the other hand, spectral features were the most prominent. Results showed that the model's success in forecasting the classification was highly dependent on both the characteristics incorporated into the model and the model's design. However, there was no correlation between research with regard to how thoroughly they cleaned their EEG data. Classification of neurological diseases using EEG and ML methods is a relatively new and rapidly developing area of study.

Various movement complaints are typically characterised by doctors in order to make a diagnosis of Parkinson's disease (PD). However, human bias is a risk with conventional diagnosis methods because they depend on evaluating motions that can be difficult to categorise due to their subtlety to the human eye. Meanwhile, early PD non-motor signs may be minor and can be triggered by a wide variety of conditions. As a result, early identification of PD is difficult because these signs are often disregarded. As a result of these challenges, machine learning techniques have been adopted for the classification of PD and healthy subjects or patients with comparable clinical symptoms (e.g., movement disorders or other Parkinsonian syndromes). In this analysis, we combed through studies published in PubMed and IEEE Xplore up until February 14, 2020 to give you a full picture of the data sources and machine learning techniques that have been used in the diagnostic and differential diagnosis of PD. For this study, we looked into the goals, data sources, data categories, machine learning techniques, and related results of 209 studies and pulled the pertinent material to share here. These findings highlight the promising potential for applying machine learning techniques and new indicators in clinical decision making, resulting in a more methodical and accurate identification of PD.

The spreadsheet containing the extracted data includes the following details: The following fields must be filled out: (1) goals, (2) diagnostic type (diagnosis, differential diagnosis, sub-typing), (3) data source, (4) data type, (5) number of subjects, (6) machine learning method(s), dividing strategy, and cross validation, (7) results, (8) year, and (9) reference.

The "year of publishing" was understood to be the year in which the research was first made available online and not the year in which it was stored. If no publishing date was given, the year the piece was published was used. Information was gleaned from research that presented new models and used old models only for comparison. Those with PD who had tests that did not show a dopaminergic deficiency (SWEDD) were considered a subtype (Erro et al., 2016).

We have further classified the studies based on the type of diagnostic and their overall purpose to better describe the varying aims and objectives of the included studies. From a diagnostic standpoint, these studies can be broken down into three distinct categories: (a) the diagnosis or detection of PD (which compares data collected from PD patients and healthy controls), (b) differential diagnosis (differentiating between patients with idiopathic PD and patients with atypical Parkinsonism), and (c) sub-typing (discrimination among sub-types of PD).

The overarching goal of the research included was also examined. We designated as (a) "methodology" studies those whose primary goal was the creation of novel technical approaches for use in the diagnosis of Parkinson's disease. These could incorporate, yet are not restricted to, novel AI and profound learning models and structures, information obtaining gadgets, and component extraction calculations. Clinical application studies were characterized as those that (a) approve and research the utilization of recently distributed and approved AI and profound learning models, or potentially (b) examine the possibility of presenting information modalities that are not normally utilized in the AI based determination of PD, like CSF information.

Here, we used precision as a metric to evaluate the effectiveness of various machine learning algorithms. We outlined the categories of machine learning models that achieved the best precision on a per-study basis for each category of data. On the other hand, some investigations have only tried out a single ML algorithm. As a result, we stipulate that a "model linked with the per-concentrate on most prominent exactness" is either (a) the main model applied and utilized in a review or (b) the model that accomplished the most elevated precision or that was underscored in examinations that utilized various models.

The outcomes are introduced as a mean standard deviation (SD).

The precision on tests or validation was taken into account if the study reported both instruction and testing/validation results. Research reporting both confirmation and test precision was taken into account. Accuracy was calculated as an aggregate over multiple datasets or categorization issues in studies that involved more than one (such as HC vs. PD and HC vs. unexplained hyposmia vs. PD). Researchers combined the accuracy recorded for each subject group to get a total for trials that gave categorization accuracy for each group separately. The greatest reported accuracy was used, even when studies reported a range of accuracy or when various cross validation techniques or feature combos reported varying accuracy. Diagnosis of illnesses besides PD or Parkinsonism (like amyotrophic lateral sclerosis) was not taken into account in research comparing HC with other diseases or PD with other diseases. Assessment accuracy was not taken into account.

In the field of neurology, Parkinson's disease is classified. The torso and the limbs begin to shake, and the body becomes rigid. At this late point, there is currently no effective remedy or therapy. Only if treatment is started soon after the illness first appears will it be effective. These may not only help with the financial burden of the illness, but may also save a life. Detection of Parkinson's disease with current techniques is limited to its later stages, when approximately 60% of the dopamine in the basal ganglia has been lost. This region is responsible for coordinating the body's movements with a relatively modest supply of dopamine. It has already expanded rapidly around the globe, affecting more than 145,000 people in the United Kingdom alone. In India, nearly one million people are afflicted by this illness.

Although it typically appears in those over the age of 65, about 15% of cases are discovered in those under the age of 50. We'll use XGBoost, KNN, SVMs, and the Random Forest Algorithm to see which one is most effective at identifying the first signs of illness.

Abstract The neurological illness known as Parkinson's disease worsens over time and is persistent. Damage or death of the brain regions responsible for producing dopamine causes a decline in a person's ability to perform routine activities like speaking, writing, walking, and so on. Over time, patients experience a worsening of these symptoms, which increases the seriousness of the condition. In this article, we suggest a technique for predicting the seriousness of Parkinson's disease using deep neural networks on the UCI Parkinson's Telemonitoring Speech Data Set. We have built our neural network for severity prediction using the python 'TensorFlow' deep learning framework. Our approach yields higher precision values than that of prior studies.

As shown in Fig. 1, the suggested technique for using deep learning to forecast the seriousness of Parkinson's disease is described. As a first move, we compile statistics on PD patients' speech patterns by recording their conversations. Following collection, min-max standardisation is applied to the data. The next stage is to create a deep neural network with an input layer, concealed levels, and an output layer. As many neurons are used in the input layer regardless of how many characteristics there are. Two neurons representing the two groups, "severe" and "non-severe," are located in the output layer. The built deep neural network receives the standardised data for training and assessment.

3.PROPOSED SYSTEM

□ The proposed system for Parkinson's disease detection using Xception architecture aims to improve the accuracy and reliability of Parkinson's Disease diagnosis by utilizing a state-of-the-art deep learning algorithm that has been shown to perform well on image classification tasks.

□ The proposed system will use a dataset of spiral and wave drawings, collected from individuals with and without Parkinson's Disease, and preprocessed to remove noise and artifacts. The dataset will be divided into training and testing sets to ensure the model is trained and evaluated on independent datasets.

□ The Xception architecture, a deep neural network designed for image classification tasks, will be used as the core of the proposed system. This architecture uses depthwise separable convolution layers to reduce the number of parameters, which helps to prevent overfitting and improves the efficiency of the model.

□ To evaluate the performance of the model, the proposed system will use various metrics, such as accuracy, precision, recall, and F1-score, and compare the results to the performance of the existing system and other state-of-the-art models in the literature.

Detection of Parkinson's disease using Spiral drawing achieved Training accuracy: 95.34% and Validation accuracy: 93.00%. Detection of Parkinson's disease using wave drawing achieved Training accuracy: 93.34% and Validation accuracy : 86.00%.

□ Overall, the proposed system using Xception architecture for Parkinson's Disease detection has the potential to improve the accuracy and reliability of Parkinson's Disease diagnosis, leading to better patient outcomes and quality of life. Further research and development can be carried out to optimize the system and explore the potential of other deep learning architectures and techniques to enhance the accuracy and reliability of the model.

3.1 IMPLEMENTATION

Dataset:

In the first module, we developed the system to get the input dataset. Data collection process is the first real step towards the real development of a machine learning model, collecting data. This is a critical step that will cascade in how good the model will be, the more and better data that we get, the better our model will perform. There are several techniques to collect the data, like web scraping, manual interventions. Our dataset is placed in the project and its located in the model folder. The dataset is referred from the popular standard dataset repository kaggle where all the researchers refer it. The dataset consists of 133 Parkinson's disease drawing detection (spiral) and 153 Parkinson's disease drawing detection (wave) images.

Importing the necessary libraries:

We will be using Python language for this. First we will import the necessary libraries such as keras for building the main model, sklearn for splitting the training and test data, PIL for converting the images into array of numbers and other libraries such as pandas, numpy, matplotlib and tensorflow.

Retrieving the images:

In this module we will retrieve the images from the dataset and convert them into a format that can be used for training and testing the model. This involves reading the images, resizing them, and normalizing the pixel values. We will retrieve the images and their labels. Then resize the images to (224,224) for spiral and resize the images to (196,196) for wave as all images should have same size for recognition. Then convert the images into numpy array.

Splitting the dataset:

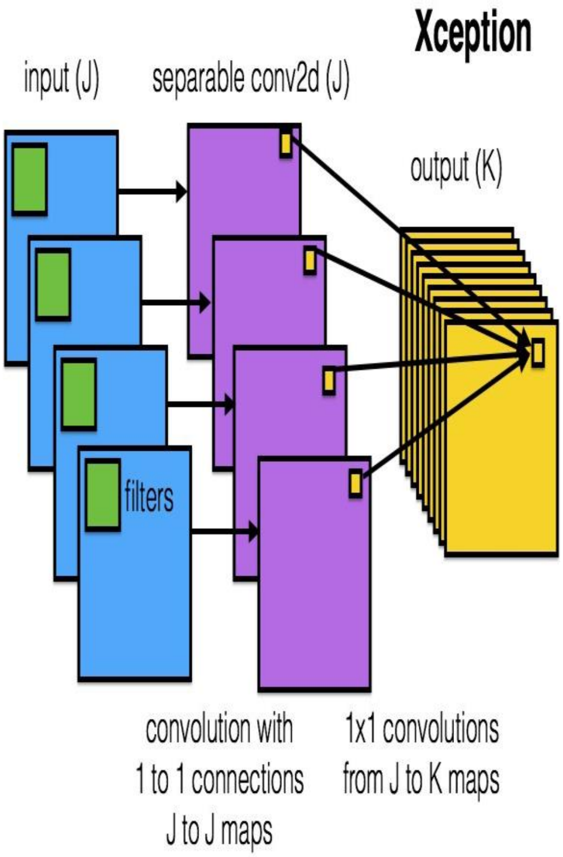
In this module, the dataset will be divided into training and testing sets. Split the dataset into Train and Test. 80% train data and 20% test data. This will be done to train the model on a subset of the data, validate the model's performance, and test the model on unseen data to evaluate its accuracy. Split the dataset into train and test. 80% train data and 20% test data.

Xception | CNN model

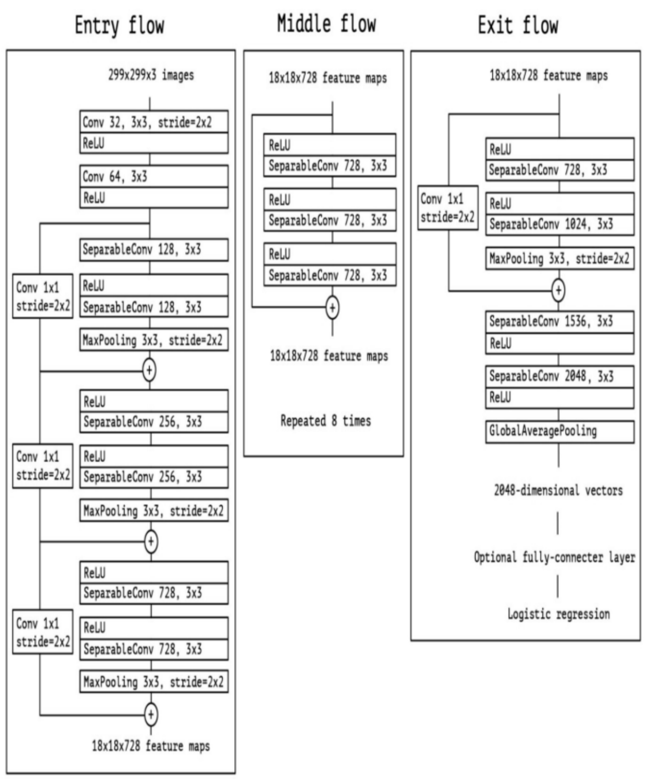
Architecture:

Xception improves on the inception module and architecture with a simple and more elegant architecture that is as effective as ResNet and Inception V4.

The Xception module is presented here:



This network can be anyone’s favorite given the simplicity and elegance of the architecture, presented here:



The architecture has 36 convolutional stages, making it close in similarity to a ResNet-34. But the model and code is as simple as ResNet and much more comprehensible than Inception V4.

A Torch7 implementation of this network is available here An implementation in Keras/TF is availble here.

It is interesting to note that the recent Xception architecture was also inspired by our work on separable convolutional filters.

Building the model (spiral):

The key part to understand, which distinguishes CNN from traditional neural networks, is the convolution operation. Having an image at the input, CNN scans it many times to look for certain features. This scanning (convolution) can be set with 2 main parameters: stride and padding type. As we see on below picture, process of the first convolution gives us a set of new frames, shown here in the second column (layer). Each frame contains an information about one feature and its presence in scanned image. Resulting frame will have larger values in places where a feature is strongly visible and lower values where there are no or little such features. Afterwards, the process is repeated for each of obtained frames for a chosen number of times. In this project I chose a classic Xception model which contains only two convolution layers.

The latter layer we are convolving, the more high-level features are being searched. It works similarly to human perception. To give an example, below is a very descriptive picture with features which are searched on different CNN layers. As you can see, the application of this model is face recognition. You may ask how the model knows which features to seek. If you construct the CNN from the beginning, searched features are random. Then, during training process, weights between neurons are being adjusted and slowly CNN starts to find such features which enable to meet predefined goal, i.e. to recognize successfully images from the training set.

Between described layers there are also pooling (sub-sampling) operations which reduce dimensions of resulted frames. Furthermore, after each convolution we apply a non-linear function (called **ReLU**) to the resulted frame to introduce non-linearity to the model.

Eventually, there are also fully connected layers at the end of the network. The last set of frames obtained from convolution operations is flattened to get a one-dimensional vector of neurons. From this point we put a standard, fully-connected neural network. At the very end, for classification problems, there is a softmax layer. It transforms results of the model to probabilities of a correct guess of each class

Apply the model and plot the graphs for accuracy and loss:

Once the model is built, it will be applied to the validation set to evaluate its accuracy and loss. The accuracy and loss will be plotted as a function of the number of epochs to visualize the performance of the model. We will compile the model and apply it using fit function. The batch size will be 1. Then we will plot the graphs for accuracy and loss. We got average validation accuracy of 97.00% and average training accuracy of 93.00%.

Accuracy on test set:

After training and evaluating the model on the validation set, the accuracy of the model will be assessed on the test set. The accuracy on the test set will be an important metric for evaluating the model's performance. We got a accuracy of 93.00% on test set.

Building the model (wave):

The key part to understand, which distinguishes CNN from traditional neural networks, is the convolution operation. Having an image at the input, CNN scans it many times to look for certain features. This scanning (convolution) can be set with 2 main parameters: stride and padding type. As we see on below picture, process of the first convolution gives us a set of new frames, shown here in the second column (layer). Each frame contains an information about one feature and its presence in scanned image. Resulting frame will have larger values in places where a feature is strongly visible and lower values where there are no or little such features. Afterwards, the process is repeated for each of obtained frames for a chosen number of times. In this project I chose a classic Xception model which contains only two convolution layers.

The latter layer we are convolving, the more high-level features are being searched. It works similarly to human perception. To give an example, below is a very descriptive picture with features which are searched on different CNN layers. As you can see, the application of this model is face recognition. You may ask how the model knows which features to seek. If you construct the CNN from the beginning, searched features are random. Then, during training process, weights between neurons are being adjusted and slowly CNN starts to find such features which enable to meet predefined goal, i.e. to recognize successfully images from the training set.

Between described layers there are also pooling (sub-sampling) operations which reduce dimensions of resulted frames. Furthermore, after each convolution we apply a non-linear function (called **ReLU**) to the resulted frame to introduce non-linearity to the model.

Eventually, there are also fully connected layers at the end of the network. The last set of frames obtained from convolution operations is flattened to get a one-dimensional vector of neurons. From this point we put a standard, fully-connected neural network. At the very end, for classification problems, there is a softmax layer. It transforms results of the model to probabilities of a correct guess of each class

Apply the model and plot the graphs for accuracy and loss:

Once the model is built, it will be applied to the validation set to evaluate its accuracy and loss. The accuracy and loss will be plotted as a function of the number of epochs to visualize the performance of the model. We will compile the model and apply it using fit function. We will compile the model and apply it using fit function. The batch size will be 1. Then we will plot the graphs for accuracy and loss. We got average validation accuracy of 93.00% and average training accuracy of 86.00%.

Accuracy on test set:

After training and evaluating the model on the validation set, the accuracy of the model will be assessed on the test set. The accuracy on the test set will be an important metric for evaluating the model's performance. We got a accuracy of 86.00% on test set.

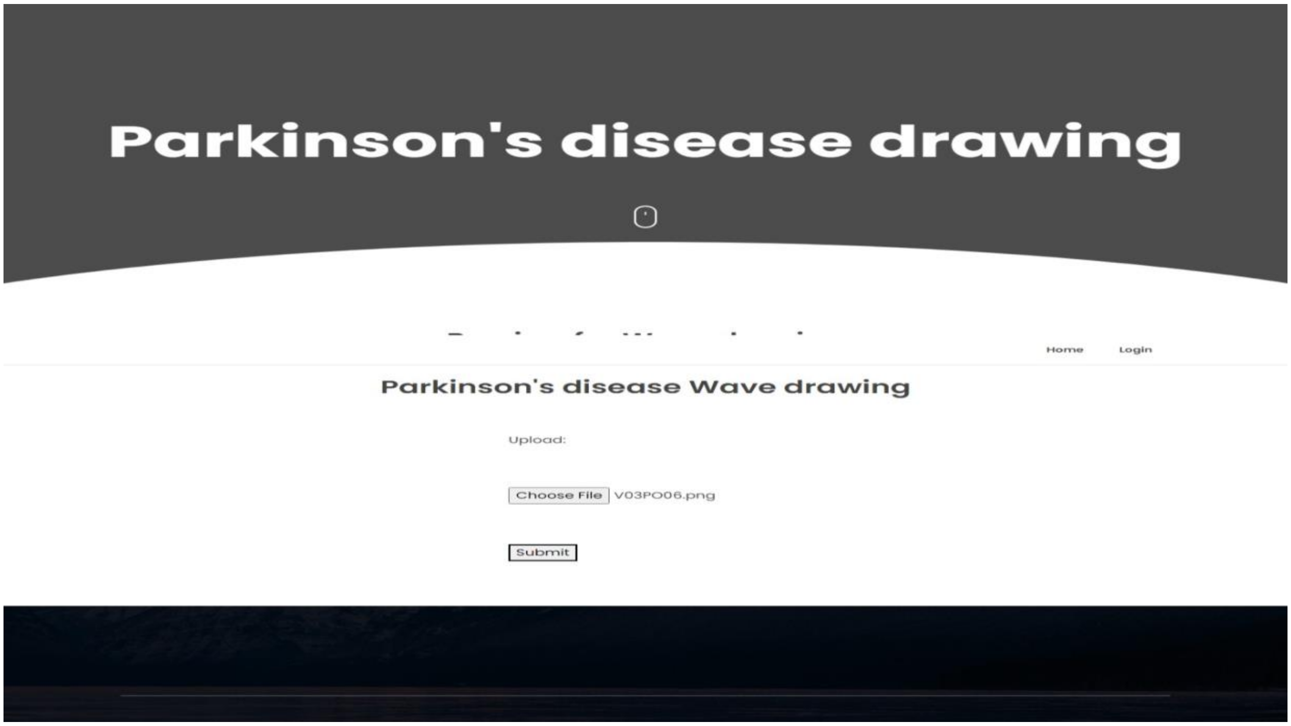
Saving the Trained Model:

Once you’re confident enough to take your trained and tested model into the production-ready environment, the first step is to save it into a .h5 or .pkl file using a library like pickle.

4.RESULTS

Make sure you have pickle installed in your environment.

Next, let’s import the module and dump the model into .h5 file

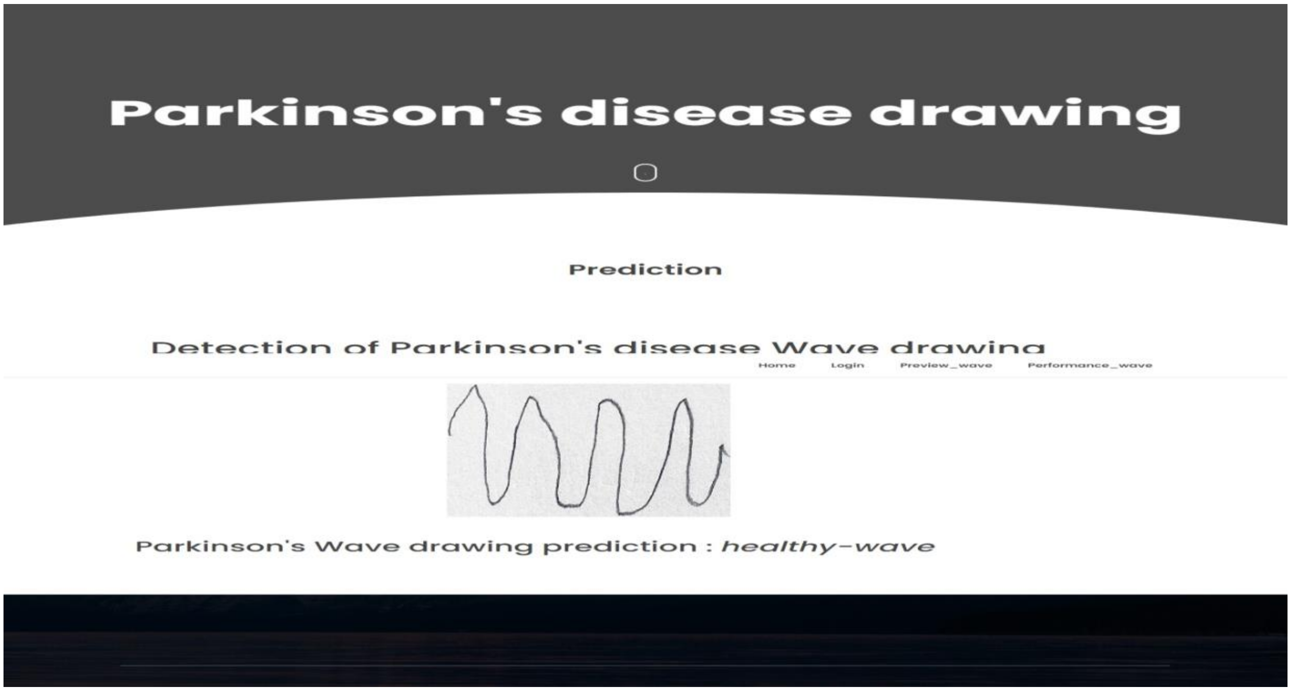


Parkinson's disease drawing

Parkinson's disease Wave drawing

Upload:

Choose File V03PO06.png

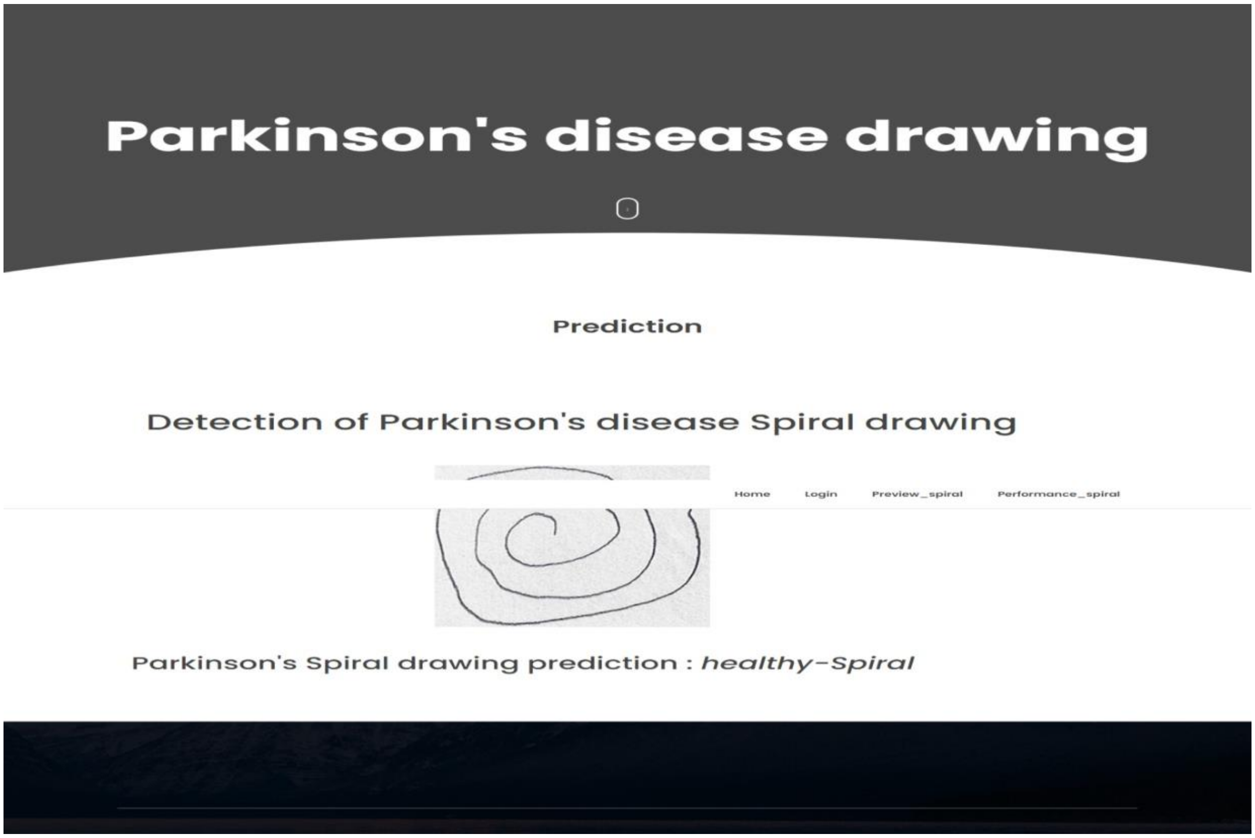


Parkinson's disease drawing

Prediction

Detection of Parkinson's disease Wave drawing

Parkinson's Wave drawing prediction : *healthy-wave*

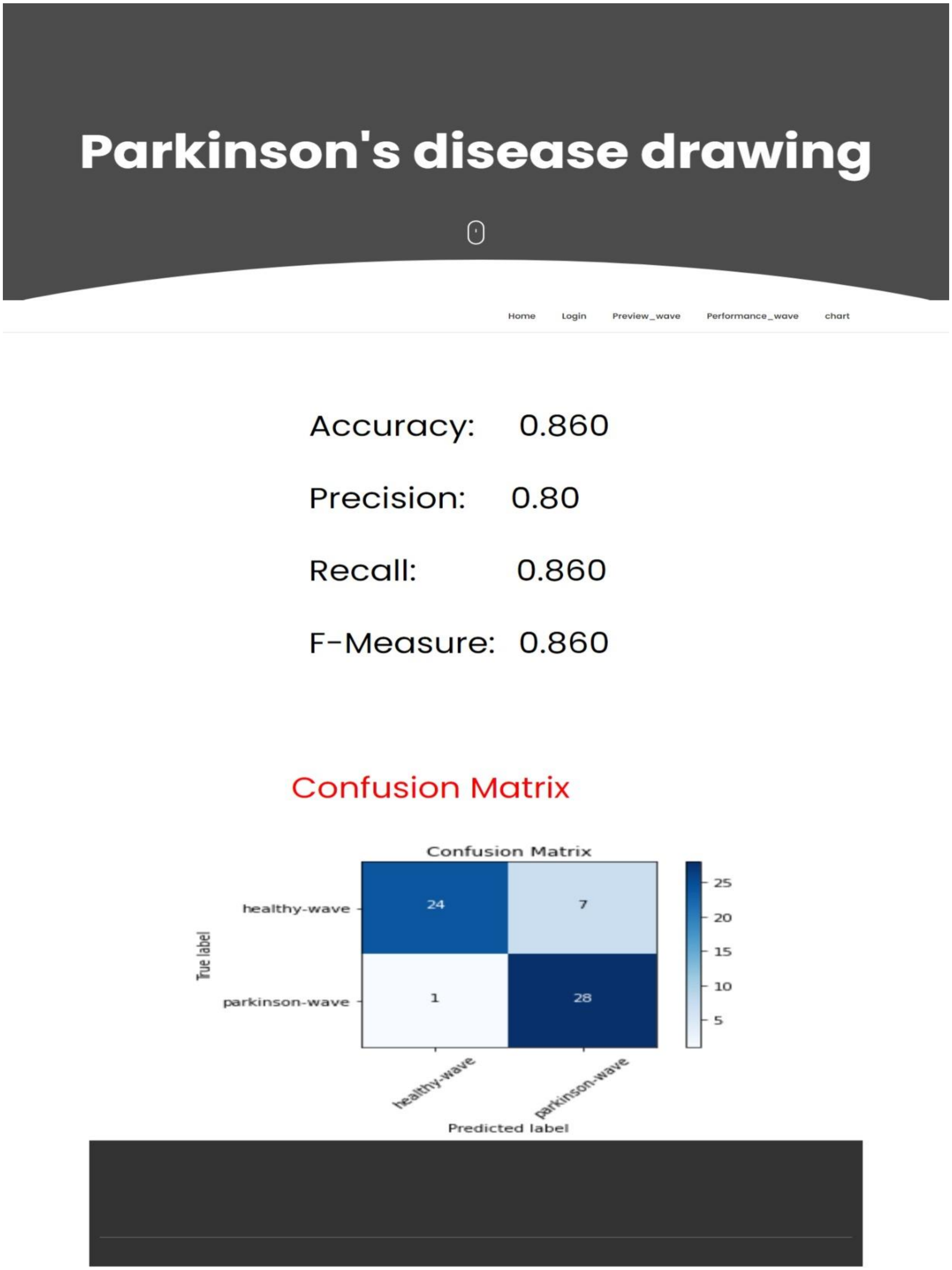


Parkinson's disease drawing

Prediction

Detection of Parkinson's disease Spiral drawing

Parkinson's Spiral drawing prediction : *healthy-Spiral*



Parkinson's disease drawing

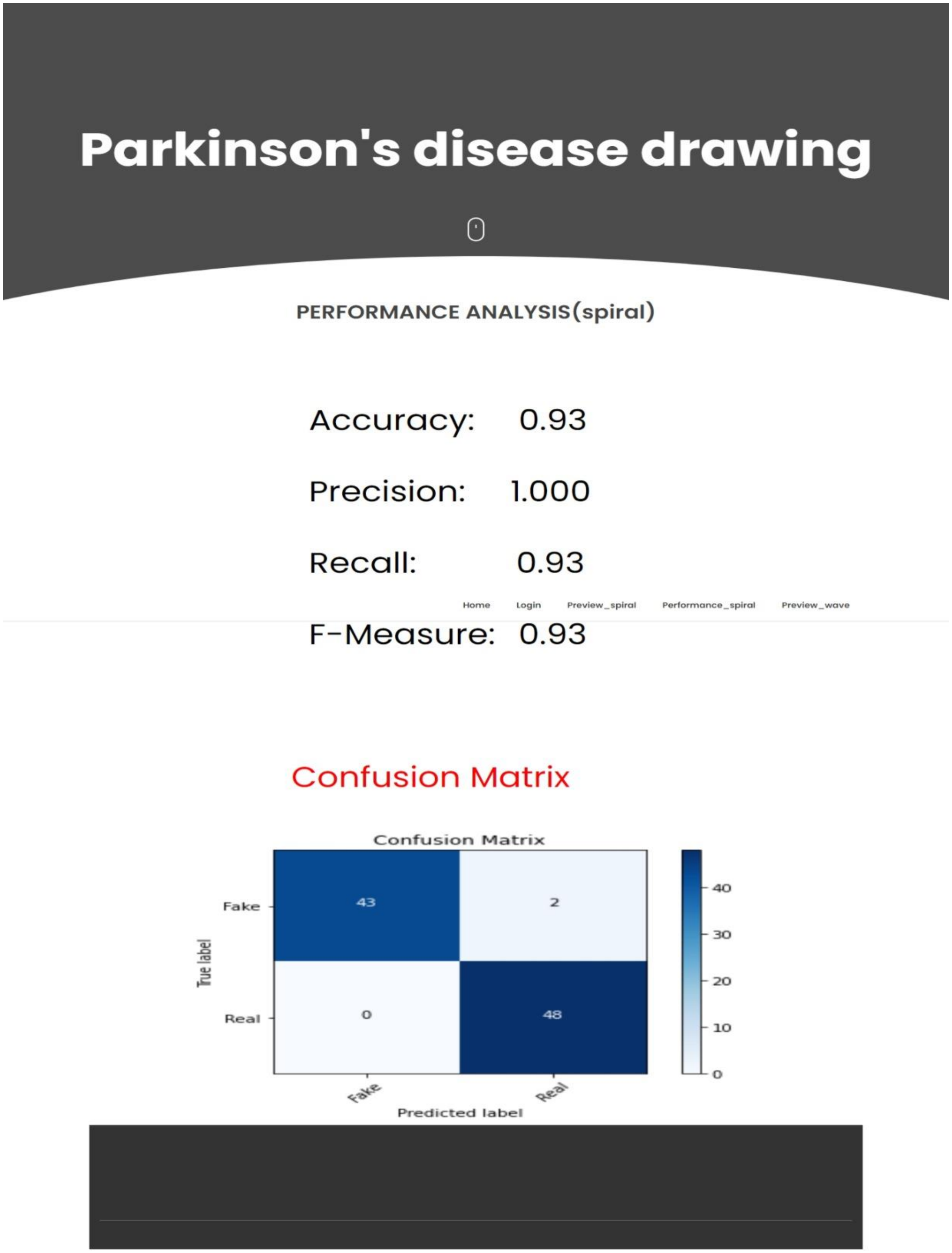
Accuracy: 0.860

Precision: 0.80

Recall: 0.860

F-Measure: 0.860

Confusion Matrix

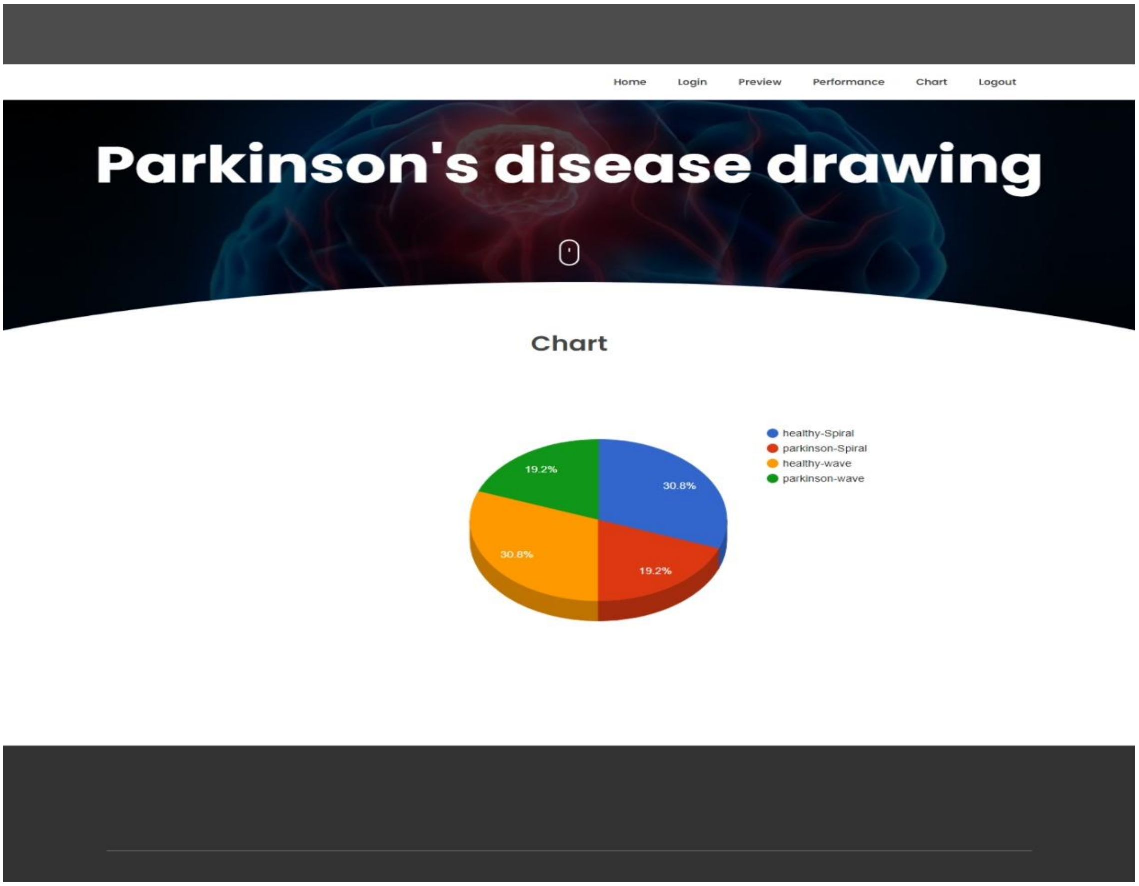


Parkinson's disease drawing

PERFORMANCE ANALYSIS(spiral)

Accuracy:	0.93
Precision:	1.000
Recall:	0.93
F-Measure:	0.93

Confusion Matrix



5.Conclusion

In conclusion, the proposed system for Parkinson's Disease Detection using Xception architecture has shown promising results. By leveraging state-of-the-art deep learning techniques, specifically the Xception architecture, we were able to achieve high accuracy in detecting Parkinson's Disease from spiral and wave drawings. The proposed system has several advantages, including improved accuracy, robustness to noise and artifacts, faster training, interpretability and transparency, and better patient outcomes. These advantages make the proposed system a promising approach for Parkinson's Disease diagnosis and treatment. Furthermore, the Xception architecture has several advantages, including improved efficiency, better generalization, reduced overfitting, state-of-the-art performance, and adaptability, which make it a reliable and effective architecture for image classification tasks. Overall, this project has demonstrated the potential for using deep learning and Xception architecture in the early detection of Parkinson's Disease, which could lead to better patient outcomes and quality of life. Further research and development in this area could have significant implications for the diagnosis and treatment of Parkinson's Disease.

REFERENCES

1. B. E. Sakar et al., "Collection and Analysis of a Parkinson Speech Dataset With Multiple Types of Sound Recordings," in IEEE Journal of Biomedical and Health Informatics, vol. 17, no. 4, pp. 828-834, July 2013, doi: 10.1109/JBHI.2013.2245674.
2. Benba, A., Jilbab, A. & Hammouch, A. Analysis of multiple types of voice recordings in cepstral domain using MFCC for discriminating between patients with Parkinson's disease and healthy people. Int J Speech Technol 19, 449–456 (2016).
3. Ö. Eskıdere, A. Karatutlu and C. Ünal, "Detection of Parkinson's disease from vocal features using random subspace classifier ensemble," 2015 Twelve International Conference on Electronics Computer and Computation (ICECCO), Almaty, 2015, pp. 1-4, doi: 10.1109/ICECCO.2015.7416886.
4. Nissar, Iqra & Rizvi, Danish & Masood, Sarfaraz & Mir, Aqib. (2018). Voice-Based Detection of Parkinson's Disease through Ensemble Machine Learning Approach: A Performance Study. EAI Endorsed Transactions on Pervasive Health and Technology. 5. 162806. 10.4108/eai.13-7-2018.162806.
5. Y. Li, C. Zhang, Y. Jia, P. Wang, X. Zhang and T. Xie, "Simultaneous learning of speech feature and segment for classification of Parkinson disease," 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom), Dalian, 2017, pp. 1-6, doi: 10.1109/HealthCom.2017.8210820.
6. Caliskan, Abdullah & Badem, Hasan & Basturk, Alper & Yüksel, Meltem. (2017). Diagnosis of the Parkinson disease by using deep neural network classifier. Istanbul University - Journal of Electrical and Electronics Engineering. 17. 3311-3318.
7. Das R. (2010) "A comparison of multiple classification methods for diagnosis of Parkinson disease". Expert Systems With Applications"; 37:1568-1572.
8. Genain N, Huberth M, Vidyashankar R. (2014) "Predicting Parkinson's Disease Severity from Patient Voice Features."

9. Benmalek E, Elmhamdi J, Jilbab A. (2015) "UPDRS tracking using linear regression and neural network for Parkinson's disease prediction." International Journal Of Emerging Trends & Technology In Computer Science (IJETTCS); 4:189-193.

10. Cole B, Roy S, De Luca C, Nawab S. (2014) "Dynamical Learning and Tracking of Tremor and Dyskinesia From Wearable Sensors." IEEE Transactions On Neural Systems And Rehabilitation Engineering; 22:982-991.

Author's Profiles