

VIRTUAL MOUSE CONTROL WITH EYEBALL MOVEMENTS USING OPENCV

A Project Report Phase – II

Submitted in partial fulfilment of the requirements for the
award of Bachelor of Engineering Degree in

Electronics and Communication Engineering

By

RAM NIVAS ANUSURI (39130030)

B.D.R.V. NAGESWARA RAO (39130064)



DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119.**

APRIL 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION
ENGINEERING
BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **RAM NIVASANUSURI (39130030)** and **B.D.R.V. NAGESWARA RAO (39130064)** whocarried out the project entitled **"VIRTUAL MOUSE CONTROL WITH EYE BALLMOVEMENTS USING OPENCV"** supervision from November 2022 toApril 2023.

Dr. D. Anbarasi Jebaselvi
15/04/23

Dr. G. D. ANBARASI JEBASELVI., M.E., Ph.D.,
Internal Guide

Dr. T. Ravi
15/4/2023

Dr. T. RAVI, M.E., Ph.D.,
Head of the Department

Submitted for Viva voce Examination held on _____

Dr. Anbarasi
19/4/23
Internal Examiner

Raj
19/4/23
External Examiner

DECLARATION

We, **RAM NIVAS ANUSURI (39130030)** and **B.D.R.V. NAGESWARA RAO (39130064)**, hereby declare that the Project Report entitled “**VIRTUAL MOUSE CONTROL WITH EYEBALL MOVEMENTS USING OPENCV**” is done under the guidance of, **Dr. G. D. ANBARASI JEBASELVI., M.E., Ph. D**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in **Electronics and Communication Engineering**.

1. 
2. 

SIGNATURE OF THE CANDIDATES

DATE: 01 - 04 -2023

PLACE: CHENNAI

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our profound gratitude to **Dr. N.M. NANDHITHA, M.E., Ph.D.**, Prof. & Dean, School of Electrical and Electronics and **Dr. T. RAVI, M.E., Ph.D.**, Head of the Department of Electronics and Communication Engineering, Sathyabama Institute of Science and Technology, or having been a constant source of support and encouragement for the completion of the project.

We consider our self-most fortunate and immensely privileged to have worked under the guidance of **Dr. G. D. ANBARASI JEBASELVI., M.E., Ph. D**, Department of Electronics and Communication Engineering to complete this project successfully.

We express our heartfelt thanks to our family members and all our friends who helped me to complete our project successfully and finally, I thank the exciting Super Natural Power, the Almighty GOD.

TABLE OF CONTENTS

CHAPTER Nos.	TITLE	PAGE Nos.
	ABSTRACT	v
	LIST OF ABBREVIATIONS	vi
	LIST OF FIGURES	vii
1.	INTRODUCTION	1
2.	LITERATURE SURVEY	19
	2.1 Hand Gesture Recognition Based Virtual Mouse Events	19
	2.2 Virtual Mouse Control Using Coloured Finger Tips and Hand Gesture Recognition	19
	2.3 Mouse on a Ring: A Mouse Action Scheme Based on IMU and Multi-Level Decision Algorithm	20
	2.4 Wireless Gyro-mouse for Text Input on a Virtual Keyboard	21
	2.5 Gesture Recognition Based Virtual Mouse and Keyboard	22
	2.6 Real-time Eye Gaze Direction Classification Using Convolutional Neural Network	22
3.	EXISTING SYSTEM	24
4.	PROPOSED SYSTEM	25
	4.1 SVM Algorithm	26
	4.2 Deep Neural Network	29
	4.3 Deep Neural Network methodology	30
	4.4 Linear Discriminator	30
5.	SOFTWARE AND ALGORITHMS USED	36
6.	RESULTS AND CONCLUSION	40
	REFERENCES	41

ABSTRACT

There are different reasons for which people need an artificial of locomotion such as a virtual keyboard. The main reason we need some alternative and innovative ideas which make our work easier. The project's main objectives are to design an accurate and efficient eye-tracking system, develop software to process the captured data, and integrate the software with the computer's operating system to allow hands-free mouse control. The system comprises of an eye tracking device and a software module. The eye-tracking device captures the user's eye movements, while the software module analyses the captured data to determine the intended mouse movements. The software module then communicates the mouse movements to the computer's operating system, allowing the user to interact with the computer hands-free. The project involves the use of machine learning algorithms and computer vision techniques to improve the accuracy of the eye-tracking system. The number of people, who need to move around with the help of some article means, because of an illness. Moreover, implementing a controlling system in it enables them to move without the help of another person is very helpful. The idea of eye controls of great use to not only the future of natural input but more importantly the handicapped and disabled. Camera will be capturing the image of eye movement. First detect pupil centre position of eye. Then the different variation on pupil position get different command set for virtual keyboard. The signals pass the motor driver to interface with the virtual keyboard itself. The motor driver will control both speed and direction to enable the virtual keyboard to move forward, left, right and stop.

LIST OF ABBREVIATIONS

HCI	Human-Computer Interface
ASR	Automatic Speech Recognition
ML	Machine Learning
SVR	Support Vector Regression
PCA	Principal Component Analysis
IE	Information Extraction
LDA	Linear Discriminant Analysis
ANOVA	Analysis of Variance
ANN	Artificial Neural Networks
SVM	Support Vector Machine
RDBMS	Relational Database Management System
LASSO	Least Absolute Shrinkage and Selection Operator
OpenCV	Open-Source Computer Vision
STT	Speech To Text
GAN	Generative Adversarial Networks

LIST OF FIGURES

FIGURE Nos.	TITLE	PAGE NOs.
4.1	Deep neural network	29
4.2	Graph of dataset	31
4.3	LDA: Projection on X-Axis	31
4.4	Dimensionality Reduction	32
4.5	Linear Discriminant Analysis in Python	32
4.6	Minimizing variance	33
4.7	LDA: Better Output	33
4.8	Block Diagram of the proposed system	34

CHAPTER 1

INTRODUCTION

As the computer technologies are growing rapidly, the importance of human computer interaction becomes highly notable. Some persons who are disabled cannot be able to use the computers. Eye ball movement control mainly used for disabled people. Incorporating this eye controlling system with the computers will make them to work without the help of other individual. Human-Computer Interface (HCI) is focused on use of computer technology to provide interface between the computer and the human. There is a need for finding the suitable technology that makes the effective communication between human and computer. Human computer interaction plays the important role. Thus, there is a need to find a method that spreads an alternate way for making communication between the human and computer to the individuals those who have impairments and give them an equivalent space to be an element of Information Society. In recent years, the human computer interfaces are attracting the attention of various researchers across the globe. Human computer interface is an implementation of the vision-based system for eye movement detection for the disabled people in the proposed system, we have included the face detection, face tracking, eye detection and interpretation of a sequence of eye blinks in real time for controlling a nonintrusive human computer interface. Conventional method of interaction with the computer with the mouse is replaced with the human eye movements. This technique will help the paralyzed person, physically challenged people especially person without hands to compute efficiently and with the ease of use. Firstly, camera captures the image and focuses on the eye in the image using OpenCV code for pupil detection. This results the centre position of the human eye (pupil). Then the centre position of the pupil is taken as a reference and based on that the human or the user will control the cursor by moving left and right. This project describes existing solutions to find the cursor movement using some 3D models. In this research work how, cursor is working based only on Eyeball movement using OpenCV methodology. And how the cursor is moving using eyeball with example with the better solutions. And, also the Conclusion part is present.

1.1 MACHINE LEARNING – OVERVIEW

Machine learning is a very hot topic for many key reasons, and because it provides the ability to automatically obtain deep insights, recognize unknown patterns, and create high performing predictive models from data, all without requiring explicit programming instructions.

This high-level understanding is critical if ever involved in a decision-making process surrounding the usage of machine learning, how it can help achieve business and project goals, which machine learning techniques to use, potential pitfalls, and how to interpret the results.

Machine learning is a subfield of computer science, but is often also referred to as predictive analytics, or predictive modelling. Its goal and usage are to build new and/or leverage existing algorithms to learn from data, in order to build generalizable models that give accurate predictions, or to find patterns, particularly with new and unseen similar data.

Imagine a dataset as a table, where the rows are each observation (aka measurement, data point, etc), and the columns for each observation represent the features of that observation and their values.

At the outset of a machine learning project, a dataset is usually split into two or three subsets. The minimum subsets are the training and test datasets, and often an optional third validation dataset is created as well.

Once these data subsets are created from the primary dataset, a predictive model or classifier is trained using the training data, and then the model's predictive accuracy is determined using the test data.

As mentioned, machine learning leverages algorithms to automatically model and find patterns in data, usually with the goal of predicting some target output or response. These algorithms are heavily based on statistics and mathematical optimization.

Optimization is the process of finding the smallest or largest value (minima or maxima) of a function, often referred to as a loss, or cost function in the

minimization case. One of the most popular optimization algorithms used in machine learning is called gradient descent, and another is known as the normal equation.

In a nutshell, machine learning is all about automatically learning a highly accurate predictive or classifier model, or finding unknown patterns in data, by leveraging learning algorithms and optimization techniques.

1.2 MACHINE LEARNING APPLICATIONS

As we move forward into the digital age, one of the modern innovations we've seen is the creation of **Machine Learning**. This incredible form of artificial intelligence is already being used in various industries and professions. For Example, Image and Speech Recognition, Medical Diagnosis, Prediction, Classification, Learning Associations, Statistical Arbitrage, Extraction, Regression. Today we're looking at all these Machine Learning Applications in today's modern world.

1.2.1 IMAGE RECOGNITION

It is one of the most common machine learning applications. There are many situations where you can classify the object as a digital image. For digital images, the measurements describe the outputs of each pixel in the image.

In the case of a **black and white image**, the intensity of each pixel serves as one measurement. So, if a black and white image has $N \times N$ pixels, the total number of pixels and hence measurement is N^2 .

In the **coloured image**, each pixel considered as providing 3 measurements of the intensities of 3 main colour components i.e. **RGB**. So, $N \times N$ coloured image there are $3 N^2$ measurements.

- **For face detection** – The categories might be face versus no face present. There might be a separate category for each person in a database of several individuals.
- **For character recognition** – It can segment a piece of writing into smaller images, each containing a single character. The categories might consist of the 26 letters of the English alphabet, the 10 digits, and some special characters.

1.2.2 Speech Recognition

Speech recognition is the translation of spoken words into text. It is also known as “automatic speech recognition” (ASR), “computer speech recognition”, or “speech to text” (STT).

In speech recognition, a software application recognizes spoken words. The measurements in this Machine Learning application might be a set of numbers that represent the speech signal. It can be segmented the signal into portions that contain distinct words or phonemes. In each segment, it will represent the speech signal by the intensities or energy in different time-frequency bands.

Although the details of signal representation are outside the scope of this program, it will represent the signal by a set of real values.

Speech recognition, Machine Learning applications include voice user interfaces. Voice user interfaces are such as voice dialling, call routing, demotic appliance control. It can also use as simple data entry, preparation of structured documents, speech-to-text processing, and plane.

1.2.3 Medical Diagnosis

Machine Learning provides methods, techniques, and tools that can help in solving diagnostic and prognostic problems in a variety of medical domains. It is being used for the analysis of the importance of clinical parameters and of their combinations for prognosis, e.g. prediction of disease progression, for the extraction of medical knowledge for outcomes research, for therapy planning and support, and for overall patient management. ML is also being used for data analysis, such as detection of regularities in the data by appropriately dealing with imperfect data, interpretation of continuous data used in the Intensive Care Unit, and for intelligent alarming resulting in effective and efficient monitoring.

It is argued that the successful implementation of ML methods can help the integration of computer-based systems in the healthcare environment providing opportunities to facilitate and enhance the work of medical experts and ultimately to improve the efficiency and quality of medical care.

In medical diagnosis, the main interest is in establishing the existence of a disease followed by its accurate identification. There is a separate category for each disease under consideration and one category for cases where no disease is present. Here, machine learning improves the accuracy of medical diagnosis by analysing data of patients.

The measurements in this Machine Learning applications are typically the results of certain medical tests (example blood pressure, temperature and various blood tests) or medical diagnostics (such as medical images), presence/absence/intensity of various symptoms and basic physical information about the patient (age, sex, weight etc). On the basis of the results of these measurements, the doctors narrow down on the disease inflicting the patient.

1.2.4 Statistical Arbitrage

In finance, statistical arbitrage refers to automated trading strategies that are typical of a short-term and involve a large number of securities. In such strategies, the user tries to implement a trading algorithm for a set of securities on the basis of quantities such as historical correlations and general economic variables. These measurements can be cast as a classification or estimation problem. The basic assumption is that prices will move towards a historical average.

We apply machine learning methods to obtain an index arbitrage strategy. In particular, it employs linear regression and support vector regression (SVR) onto the prices of an exchange-traded fund and a stream of stocks. By using principal component analysis (PCA) in reducing the dimension of feature space, it observes the benefit and note the issues in the application of SVR. To generate trading signals model the residuals from the previous regression as a mean reverting process.

In the case of classification, the categories might be sold, buy or do nothing for each security. In the case of estimation one might try to predict the expected return of each security over a future time horizon. In this case, one typically needs to use the estimates of the expected return to make a trading decision (buy, sell, etc.)

1.2.5 Learning Associations

Learning association is the process of developing insights into various associations between products. A good example is how seemingly unrelated products may reveal an association to one another. When analysed in relation to buying behaviours of customers.

One application of machine learning- Often studying the association between the products people buy, which is also known as basket analysis. If a buyer buys 'X', would he or she force to buy 'Y' because of a relationship that can identify between them? This leads to the relationship that exists between fish and chips etc. when new products launch in the market a Knowing these relationships it develops a new relationship. Knowing these relationships could help in suggesting the associated product to the customer. For a higher likelihood of the customer buying it, it can also help in bundling products for a better package.

This learning of associations between products by a machine is learning associations. Once we found an association by examining a large amount of sales data, Big Data analysts. It can develop a rule to derive a probability test in learning a conditional probability.

1.2.6 Classification

Classification is a process of placing each individual from the population under study in many classes. This is identified as independent variables.

Classification helps analysts to use measurements of an object to identify the category to which that object belongs. To establish an efficient rule, analysts use data. Data consists of many examples of objects with their correct classification.

For example, before a bank decides to disburse a loan, it assesses customers on their ability to repay the loan. By considering factors such as customer's earning, age, savings and financial history we can do it. This information is taken from the past data of the loan. Hence, Seeker uses to create a relationship between customer attributes and related risks.

1.2.7 Prediction

Consider the example of a bank computing the probability of any of loan applicants faulting the loan repayment. To compute the probability of the fault, the system will first need to classify the available data in certain groups. It is described by a set of rules prescribed by the analysts.

Once we do the classification, as per need we can compute the probability. These probability computations can compute across all sectors for varied purposes

The current prediction is one of the hottest machine learning algorithms. Let's take an example of retail, earlier we were able to get insights like sales report last month / year / 5-years / Diwali / Christmas. This type of reporting is called as historical reporting. But currently business is more interested in finding out what will be my sales next month / year / Diwali, etc. so that business can take a required decision (related to procurement, stocks, etc. on time.

1.2.8 Extraction

Information Extraction (IE) is another application of machine learning. It is the process of extracting structured information from unstructured data. For example, web pages, articles, blogs, business reports, and e-mails. The relational database maintains the output produced by the information extraction.

The process of extraction takes input as a set of documents and produces a structured data. This output is in a summarized form such as an excel sheet and table in a relational database.

Nowadays extraction is becoming a key in the big data industry.

As we know that the huge volume of data is getting generated out of which most of the data is unstructured. The first key challenge is handling unstructured data. Now conversion of unstructured data to structured form based on some pattern so that the same can stored in RDBMS.

Apart from this, in current days data collection mechanism is also getting change. Earlier we collected data in batches like End-of-Day (EOD), but now business wants the data as soon as it is getting generated, i.e. in real time.

1.2.9 Regression

We can apply Machine learning to regression as well. Assume that $x = x_1, x_2, x_3, \dots, x_n$ are the input variables and y is the outcome variable. In this case, we can use machine learning technology to produce the output (y) on the basis of the input variables (x). You can use a model to express the relationship between various parameters as below in equation 1:

$$Y = g(x) \quad (1)$$

where g is a function that depends on specific characteristics of the model. In regression, we can use the principle of machine learning to optimize the parameters. To cut the approximation error and calculate the closest possible outcome.

We can also use Machine learning for function optimization. We can choose to alter the inputs to get a better model. This gives a new and improved model to work with. This is known as response surface design.

So, this was all about Machine Learning Applications. Hope you like our explanation.

1.2 MACHINE LEARNING – TASKS

The **most common machine learning tasks** that one may come across while trying to solve a machine learning problem. Under each task are also listed a set of **machine learning methods** that could be used to resolve these tasks. Please feel free to comment/suggest if I missed mentioning one or more important points. Also, sorry for the typos.

Following are the key machine learning tasks briefed later in this article:

- Feature selection
- Regression
- Classification
- Clustering
- Multivariate querying
- Density estimation
- Dimension reduction
- Testing and matching

Following are top 8 most common machine learning tasks that one could come across most frequently while solving an advanced analytics problem:

Feature Selection: Feature selection is one of the critical tasks which would be used when building machine learning models. Feature selection is important because selecting right features would not only help build models of higher accuracy but also help achieve objectives related to building simpler models, reduce overfitting etc. The following are some of the techniques which could be used for feature selection:

Filter methods which helps in selecting features based on the outcomes of statistical tests. The following are some of the statistical tests which are used:

- Pearson's correlation
- Linear discriminant analysis (LDA)
- Analysis of Variance (ANOVA)
- Chi-square tests

Wrapper methods which help in feature selection by using a subset of features and determining the model accuracy. The following are some of the algorithms used:

- Forward selection
- Backward elimination
- Recursive feature elimination

Regularization techniques which penalize one or more features appropriately to come up with most important features. The following are some of the algorithms used:

- LASSO (L1) regularization
- Ridge (L2) regularization

Regression

Regression tasks mainly deal with estimation of numerical values (**continuous variables**). Some of the examples include estimation of housing price, product price, stock price etc. Some of the following ML methods could be used for solving regressions problems:

- Kernel regression (Higher accuracy)
- Gaussian process regression (Higher accuracy)
- Regression trees
- Linear regression
- Support vector regression
- LASSO

Classification

Classification tasks is simply related with predicting a category of a data (**discrete variables**). One of the most common examples is predicting whether or not an email is spam or ham. Some of the common use cases could be found in the area of healthcare such as whether a person is suffering from a particular disease or not. It also has its application in financial use cases such as determining whether

a transaction is fraud or not. The ML methods such as following could be applied to solve classification tasks:

- Kernel discriminant analysis (Higher accuracy)
- K-Nearest Neighbours (Higher accuracy)
- Artificial neural networks (ANN) (Higher accuracy)
- Support vector machine (SVM) (Higher accuracy)
- Random forests (Higher accuracy)
- Decision trees
- Boosted trees
- Logistic regression
- naive Bayes
- Deep learning

Clustering: Clustering tasks are all about finding natural groupings of data and a label associated with each of these groupings (clusters). Some of the common example includes customer segmentation, product features identification for product roadmap. Some of the following are common ML methods:

- Mean-shift (Higher accuracy)
- Hierarchical clustering
- K-means
- Topic models

Multivariate querying: Multivariate querying is about querying or finding similar objects. Some of the following ML methods could be used for such problems:

- Nearest neighbours
- Range search
- Farthest neighbours

Density estimation: Density estimation problems are related with finding likelihood or frequency of objects. In probability and statistics, density estimation is the construction of an estimate, based on observed data, of an unobservable underlying

probability density function. Some of the following ML methods could be used for solving density estimation tasks:

- Kernel density estimation (Higher accuracy)
- Mixture of Gaussians
- Density estimation tree

Dimension reduction: Dimension reduction is the process of reducing the number of random variables under consideration, and can be divided into feature selection and feature extraction. Following are some of ML methods that could be used for dimension reduction:

- Manifold learning/KPCA (Higher accuracy)
- Principal component analysis
- Independent component analysis
- Gaussian graphical models
- Non-negative matrix factorization
- Compressed sensing

Testing and matching: Testing and matching tasks relates to comparing data sets. Following are some of the methods that could be used for such kind of problems:

- Minimum spanning tree
- Bipartite cross-matching
- N-point correlation

1.4 MACHINE LEARNING SYSTEM CLASSIFICATION

Machine learning systems can be classified into several categories based on different criteria. Here are some common ways to classify machine learning systems:

1. **Supervised Learning:** In supervised learning, the machine learning system is trained using labelled data, which means that the input data is accompanied by the correct output or label. The goal of supervised learning is to learn a mapping function from input to output that can generalize well to new, unseen data. Examples of supervised learning algorithms include regression, classification, and neural networks.
2. **Unsupervised Learning:** In unsupervised learning, the machine learning system is trained using unlabelled data, which means that the input data is not accompanied by any labels or targets. The goal of unsupervised learning is to learn the underlying structure or patterns in the data, such as clustering or dimensionality reduction. Examples of unsupervised learning algorithms include k-means clustering, principal component analysis (PCA), and autoencoders.
3. **Semi-Supervised Learning:** In semi-supervised learning, the machine learning system is trained using both labelled and unlabelled data. The goal of semi-supervised learning is to leverage the unlabelled data to improve the performance of the supervised learning model. Examples of semi-supervised learning algorithms include self-training, co-training, and multi-view learning.
4. **Reinforcement Learning:** In reinforcement learning, the machine learning system learns by interacting with an environment and receiving feedback in the form of rewards or punishments. The goal of reinforcement learning is to learn a policy or strategy that maximizes the expected cumulative reward over time. Examples of reinforcement learning algorithms include Q-learning, policy gradient methods, and actor-critic methods.
5. **Online Learning:** In online learning, the machine learning system learns from a stream of data that arrives in real-time. The goal of online learning is to update the model parameters incrementally as new data arrives, without having to retrain the model from scratch. Examples of online learning

algorithms include stochastic gradient descent (SGD), online gradient descent, and perceptron learning.

These are some of the common ways to classify machine learning systems. However, there are other criteria such as model complexity, scalability, and interpretability that can also be used to categorize different machine learning algorithms.

Supervised Learning

These algorithms are trained using labelled examples, in different scenarios, as an input where the desired outcome is already known. An equipment, for instance, could have data points such as "F" and "R" where "F" represents "failed" and "R" represents "runs". A learning algorithm will receive a set of input instructions along with the corresponding accurate outcomes. The learning algorithm will then compare the actual outcome with the accurate outcome and flag an error, if there is any discrepancy. Using different methods, such as regression, classification, gradient boosting, and prediction, supervised learning uses different patterns to proactively predict the values of a label on extra unlabelled data. This method is commonly used in areas where historical data is used to predict events that are likely to occur in the future. For instance, anticipate when a credit card transaction is likely to be fraudulent or predict which insurance customers are likely to file their claims.

There are several techniques in supervised learning, including:

Regression: Regression is a technique used in supervised learning for predicting continuous target variables. Examples include predicting house prices or stock prices.

Classification: Classification is a technique used in supervised learning for predicting discrete target variables. Examples include classifying images into different categories or predicting whether a customer will churn or not.

Decision Trees: Decision trees are a type of supervised learning algorithm used for both classification and regression tasks. Decision trees recursively partition the

feature space into smaller regions, where each region corresponds to a different class or target value.

Random Forests: Random forests are an ensemble learning technique used in supervised learning for classification and regression tasks. A random forest is composed of multiple decision trees, each trained on a different subset of the data.

Support Vector Machines (SVMs): SVMs are a popular supervised learning algorithm used for classification and regression tasks. SVMs find the optimal hyperplane that separates the data into different classes or predicts the target value.

Artificial Neural Networks (ANNs): ANNs are a type of supervised learning algorithm inspired by the structure and function of biological neurons. ANNs are used for a variety of tasks, including image classification, speech recognition, and natural language processing.

Gradient Boosting: Gradient boosting is a machine learning technique that uses an ensemble of weak models to make accurate predictions. In gradient boosting, each new model is trained to correct the errors of the previous model, which improves the overall performance of the model.

Unsupervised Learning

This method of ML finds its application in areas where data has no historical labels. Here, the system will not be provided with the "right answer" and the algorithm should identify what is being shown. The main aim here is to analyse the data and identify a pattern and structure within the available data set. Transactional data serves as a good source of data set for unsupervised learning.

For instance, this type of learning identifies customer segments with similar attributes and then lets the business to treat them similarly in marketing campaigns. Similarly, it can also identify attributes that differentiate customer segments from one another. Either ways, it is about identifying a similar structure in the available data set. Besides, these algorithms can also identify outliers in the available data sets.

Some of the techniques used in Unsupervised Learning:

Clustering: Clustering is a technique used to group similar data points together based on their attributes or features. The goal of clustering is to identify natural groupings or clusters in the data without any prior knowledge of the class labels. There are various clustering algorithms such as k-means, hierarchical clustering, and density-based clustering.

Dimensionality Reduction: Dimensionality reduction is a technique used to reduce the number of variables or features in the data while retaining as much information as possible. The goal of dimensionality reduction is to simplify the data and remove any noise or redundant information that may hinder the performance of machine learning models. There are various dimensionality reduction techniques such as principal component analysis (PCA), t-SNE, and autoencoders.

Anomaly Detection: Anomaly detection is a technique used to identify data points that are significantly different from the rest of the data. The goal of anomaly detection is to identify unusual or rare events that may indicate fraud, errors, or anomalies in the data. There are various anomaly detection techniques such as density-based methods, distance-based methods, and machine learning-based methods.

Association Rule Mining: Association rule mining is a technique used to identify frequent patterns or associations between items in a dataset. The goal of association rule mining is to discover hidden relationships or dependencies between variables that may not be obvious from the data. There are various association rule mining algorithms such as Apriori, FP-Growth, and Eclat.

Generative Models: Generative models are a class of unsupervised learning algorithms that learn to generate new data that is similar to the original data. The goal of generative models is to learn the underlying distribution of the data and generate new samples that are indistinguishable from the original data. There are various generative models such as autoencoders, variational autoencoders, and generative adversarial networks (GANs).

Semi-supervised Learning

Semi-supervised learning is a type of machine learning where the machine learning system is trained using both labelled and unlabelled data. The goal of semi-supervised learning is to leverage the unlabelled data to improve the performance of the supervised learning model.

In a typical supervised learning scenario, the machine learning system is trained using labelled data, where each data point is associated with a specific label or target variable. The model learns to map the input features to the output labels using the training data. However, labelled data is often scarce and expensive to obtain, which limits the performance of supervised learning models.

In semi-supervised learning, the machine learning system is trained using both labelled and unlabelled data. The unlabelled data is used to learn the underlying structure or patterns in the data, which can then be used to improve the performance of the supervised learning model. This is done by incorporating the information from the unlabelled data into the learning process in various ways.

One common approach to semi-supervised learning is to use self-training, where the machine learning system trains on the labelled data initially and then uses the trained model to make predictions on the unlabelled data. The predicted labels are then used as if they were true labels for the unlabelled data, and the model is retrained using both the labelled and the pseudo-labelled data.

Another approach is to use co-training, where the machine learning system trains on two different sets of features or views of the data. Each view is initially trained using the labelled data, and the two views are then combined to make predictions on the unlabelled data. The predicted labels from each view are used to train the other view, which in turn improves the performance of the combined model.

Semi-supervised learning is particularly useful in scenarios where labelled data is scarce, but unlabelled data is abundant. This is often the case in many real-world applications such as speech recognition, image classification, and natural language processing. By leveraging the unlabelled data, semi-supervised learning can

improve the performance of supervised learning models without the need for additional labelled data.

Reinforcement Learning

Reinforcement learning is a type of machine learning where the machine learning system learns by interacting with an environment through a series of actions and feedback in the form of rewards or penalties. The goal of reinforcement learning is to learn a policy or a set of actions that maximizes the cumulative reward over time.

The reinforcement learning process can be formalized using the Markov decision process (MDP) framework. An MDP is defined by a set of states, a set of actions, a transition function that specifies the probability of moving from one state to another when an action is taken, and a reward function that specifies the reward received for each state-action pair. The reinforcement learning process can be summarized as follows:

- The agent observes the current state of the environment.
- The agent selects an action based on its current policy or strategy.
- The environment transitions to a new state based on the selected action.
- The agent receives a reward signal from the environment based on the new state and the selected action.
- The agent updates its policy or strategy based on the observed reward and the new state.

The goal of the agent is to learn a policy or a set of actions that maximizes the expected cumulative reward over time. This is often done using techniques such as Q-learning, policy gradients, and actor-critic methods.

Reinforcement learning has been successfully applied in various domains, such as robotics, game playing, and recommendation systems. One of the notable applications of reinforcement learning is the development of AlphaGo, an AI program that defeated the world champion in the ancient board game Go.

CHAPTER 2

LITERATURE SURVEY

2.1 Hand Gesture Recognition Based Virtual Mouse Events

Manav Ranawat, Madhur Rajadhyaksha, Neha Lakhani, Radha Shankarmani (2021) determined a virtual mouse application based on the tracking of different hand gestures. The system eliminates the dependency on any external hardware required to perform mouse actions. A built-in camera tracks the user's hands, predefined gestures are recognized and the corresponding mouse events are executed. This system has been implemented in Python using OpenCV and PyAutoGUI. Researchers have studied background conditions, effects of differences in illuminance and skin colour individually. However, the proposed system aims to consider all the above factors to build an application most suitable in the real world.

2.2 Virtual Mouse Control Using Coloured Finger Tips and Hand Gesture Recognition

Vantukala VishnuTeja Reddy, Thumma Dhyanchand, Galla Vamsi Krishna, Satish Maheshwaram (2022) determined that in human-computer interaction, virtual mouse implemented with fingertip recognition and hand gesture tracking based on image in a live video is one of the studies. In this paper, virtual mouse control using fingertip identification and hand gesture recognition is proposed. This study consists of two methods for tracking the fingers, one is by using coloured caps and other is by hand gesture detection. This includes three main steps that are finger detection using colour identification, hand gesture tracking and implementation on on-screen cursor. In this study, hand gesture tracking is generated through the detection of the contour and formation of a convex hull around it. Features of hands are extracted with the area ratio of contour and hull formed. Detailed tests are performed to check this algorithm in real world scenarios. A method for on-screen cursor control without any physical connection to a sensor is presented. Identification of coloured caps on the fingertips and their tracking is involved in this work. Different hand gestures can be replaced in place of coloured caps for the same purpose. Different operations of

mouse controlled are single left click, double left click, right click and scrolling. Various combinations of the coloured caps are used for different operations. Range of skin colour can be varied in the program in accordance with the person to be used, surrounding lightening conditions. An approximate area ratio that is not being used by the hand in the convex hull is taken after analysing the program output at different gestures of the hand. This work can be used in various real time applications like cursor control in a computer, android based smart televisions etc. Although there are devices like mouse and laser remotes for the same purpose, this work is so simple so that it reduces the usage of external hardware in such a way that the motion of fingers in front of a camera will result in the necessary operation on the screen.

2.3 Mouse on a Ring: A Mouse Action Scheme Based on IMU and Multi-Level Decision Algorithm

Yuliang Zhao (2022) determined that the traditional mouse has been used as a main tool for human-computer interaction for more than 50 years. However, it has become unable to cater to people's need for mobile officing and all-weather use due to its reliance on the support of a two-dimensional plane, poor portability, wearisomeness, and other problems. In this paper, they proposed a portable ring-type wireless mouse scheme based on IMU sensors and a multi-level decision algorithm. The user only needs to operate in the air with a smart ring worn on the middle finger of their right hand to realize the interactive function of a mouse. The smart ring first captures changes in the finger's attitude angle to reflect how the cursor position changes. And, it captures the rapid rotation of the user's palm to the left and right to achieve mouse clicking. In addition, a multi-level decision algorithm is developed to improve the response speed and recognition accuracy of the virtual mouse. The experimental results show that the virtual mouse has a target selection accuracy of over 96%, which proves its practicability in real-world applications. This virtual mouse is expected to be used as a portable and reliable tool for multi-scenario human-computer interaction applications in the future. In this paper, they proposed a ring-type wireless mouse scheme based on IMU sensors and a multi-level decision algorithm. This scheme does not require the support of a 2D desktop. The

user can realize the interactive function of a traditional mouse in the air by simply wearing a smart ring on the middle finger of their right hand. First, the smart ring captures changes in the finger's attitude angle to control how the cursor moves. This reduces the dependence on the environment and allows for unrestricted interaction in multiple scenarios. Second, the smart ring, which is integrated on the ring carrier without extra auxiliary sensors, captures the rapid rotation of the user's palm to the left and right to achieve mouse clicking. This design greatly reduces the virtual mouse's size and improves its portability. Finally, considering that the increase in user motion types affects the recognition performance, a multilevel decision algorithm is developed based on the hierarchy in the data set and used to improve the response speed and recognition accuracy of the virtual mouse. The experimental results show that the target selection accuracy of the virtual mouse is over 96%, and its target selection time and path efficiency are very close to those of a standard mouse. These findings demonstrate the effectiveness and usability of the virtual mouse in real-world applications. They believe that this virtual mouse will provide a portable, highly accurate, and responsive solution to support the needs for multi-scenario human-computer interaction in the future.

2.4 Wireless Gyro-mouse for Text Input on a Virtual Keyboard

Rares Pogoreanu and Radu Gabriel Bozomitu (2022) determined that, a gyroscopic pointing device that allows a user to type text by using a specialized virtual keyboard. They compared different typing methods that use the dwell time of the pointer as a character selection method. The proposed solution can be used in different applications, such as: computer gaming, virtual reality, remote control and to facilitate communication with people affected by certain disabilities and can easily accommodate more types of users. The performances of different methods have been analysed by using the speed of typing measured in words per minute, the task success rate, as well as the feedback from the user regarding comfort and perceived ease-of-use. In this paper, a robust method of typing based on a wireless gyro-mouse has been presented. The proposed interface includes a specialized virtual keyboard with dwell time as a selection method and a wireless gyro-mouse based on an IMU sensor that combines an accelerometer and gyroscope in a single package. The proposed device has been tested on 10 healthy subjects, students of

“Gheorghe Asachi” Technical University of Iasi, Romania. According to the obtained results, all subjects succeeded in using the device as intended, and obtained good performances in the typing test. All subjects rapidly and easily learned to use the device and succeed in completing the required tasks. The best performance results have been obtained for a dwell time of 750 ms, where the average speed of typing for all subjects was 4.1WPM.

2.5 Gesture Recognition Based Virtual Mouse and Keyboard

Nowadays computer vision has reached its pinnacle, where a computer can identify its owner using a simple program of image processing. In this stage of development, people are using this vision in many aspects of day to day life, like Face Recognition, Colour detection, Automatic car, etc. In this project, computer vision is used in creating an Optical mouse and keyboard using hand gestures. The camera of the computer will read the image of different gestures performed by a person's hand and according to the movement of the gestures the Mouse or the cursor of the computer will move, even perform right and left clicks using different gestures. Similarly, the keyboard functions may be used with some different gestures, like using one finger gesture for alphabet select and four-figure gesture to swipe left and right.

2.6 Real-time Eye Gaze Direction Classification Using Convolutional Neural Network

Anjith George, Aurobinda Routray (2016) determine the estimation eye gaze direction is useful in various human-computer interaction tasks. Knowledge of gaze direction can give valuable information regarding users point of attention. Certain patterns of eye movements known as eye accessing cues are reported to be related to the cognitive processes in the human brain. They proposed a real-time framework for the classification of eye gaze direction and estimation of eye accessing cues. In the first stage, the algorithm detects faces using a modified version of the Viola-Jones algorithm. A rough eye region is obtained using geometric relations and facial landmarks. The eye region obtained is used in the subsequent stage to classify the eye gaze direction. A convolutional neural network is employed in this work for the

classification of eye gaze direction. The proposed algorithm was tested on Eye Chimera database and found to outperform state of the art methods. The computational complexity of the algorithm is very less in the testing phase. The algorithm achieved an average frame rate of 24 fps in the desktop environment. In this work, a framework for real-time classification of eye gaze direction is presented. The estimated eye gaze direction is used to infer eye accessing cues, giving information about the cognitive states. The computational complexity is very less; They achieved frame rates around 24 Hz in Python implementation in a 2.0 GHz Core i5 PC running Ubuntu 64 bit (4GB RAM). The per-frame computational time is 42 ms, which is much less than that of the other state of the art methods (250 ms in [20]). Off the shelf webcams can be used for computing the Eye gaze direction. The proposed algorithm works even with in plane rotations of the face. The eye gaze direction obtained can also be used for human-computer interaction applications. The computational complexity of the algorithm in testing phase is less, which makes it suitable for smart devices with low resolution cameras using pre-trained models.

CHAPTER 3

EXISTING SYSTEM

MATLAB detect the iris and control cursor. Eye movement-controlled wheel chair is existing one that controls the wheel chair by monitoring eye movement. In MATLAB is difficult to predict the Centroid of eye so they go for OpenCV. The different eye-motions are classified with the help of support vector machine classifier. The eye-movements are eye open, eye close, eyeball left and eyeball right are captured by web camera. SVM can analyze data and used for classification and regression analysis. SVM is a set of associated supervised learning functions used for classification and regression problems. In SVM, the multi class file is used. This PIR sensor is connected to the General Purpose I/O port of the Raspbian board, it detects the eye pupil movement and hence makes the camera to start capturing the images. Sensor can cover the range up to 5cm. The existing eye- tracking HMD system (ET-HMD) is designed based on the rotational symmetric system. The tracking principle in this paper is based on pupil-corneal reflection.

The ET-HMD system comprises three optical paths for virtual display, infrared illumination, and eye tracking. The display optics is shared by three optical paths and consists of four spherical lenses. For the eye-tracking path, an extra imaging lens is added to match the image sensor and achieve eye tracking. Existing eye-tracking technology has been used within a multitude of disciplines to provide data linking eye movements to visual processing of various stimuli (i.e., x-rays, situational positioning, printed information, and warnings). Existing eye-tracking system for non-invasive recording of eye movements. The eye tracker uses a flying-spot laser to selectively image landmarks on the eye and, subsequently, measure horizontal, vertical, and torsional eye movements. Considerable work was required to overcome the adverse effects of specular reflection of the flying-spot from the surface of the eye onto the sensing elements of the eye tracker. These effects have been largely overcome, and the eye-tracker has been used to document eye movement abnormalities, such as abnormal torsional pulsion of saccades, in the clinical setting.

CHAPTER 4

PROPOSED SYSTEM

Utilizing both vertical integral projection and horizontal projection, it is possible to determine the pupil's precise location. The entire image is divided into homogenous subsets by these projections. The proposed method makes use of an arbitrary threshold. A Gaussian filter can be used to eliminate noise. The minimum gradient point is used to determine the strong pixel value. In the contrast region, the lower threshold guards against splitting edges. In the proposed system, MediaPipe library can be used. MediaPipe is an open source, cross-platform, customizable ML solution for live and streaming media.

In this proposed work, it can be done using OpenCV library to capture the face and send to the MediaPipe library. OpenCV (Open Source Computer Vision Library) is a Python library, is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code. It uses VideoCapture function from the OpenCV library to capture the video continuously through the laptop camera. The captured frames are sent to the MediaPipe which is a Machine Learning library to predict the face and marks the different parts of the eye.

In this proposed work, MediaPipe is used to offer ready-to-use yet customizable Python solutions as a prebuilt Python package. MediaPipe Python package is available on PyPI for Linux, macOS and Windows. Tip: Use command deactivate to later exit the Python virtual environment.

MediaPipe make use of SVM algorithm to detect face. The MediaPipe Face Detector task lets you detect faces in an image or video. You can use this task to locate faces within a frame and landmark facial features. This task uses a machine learning (ML) model that can work with single images or a continuous stream of images. The task outputs face locations, along with 6 face landmarks, including

eyes, ears, nose, and mouth. MediaPipe marks different points on the face and detects the different parts of the face. We use FaceMesh function in the MediaPipe library to mark landmarks on the face. After marking different landmarks on the face, We made some calculation that can detect when an is blink and detects the movement of the black eye ball.

4.1 SVM Algorithm

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems.

There are two types of SVM:

1. Linear SVM
2. Non-Linear SVM

Linear SVM

If a set of data can be split into two classes by just one straight line, it is said to be linearly separated, and the classification algorithm used for it is referred to as a Linear SVM.

Non - Linear SVM

A support vector machine (SVM) is a popular supervised machine learning algorithm that can be used for classification or regression tasks. A non-linear SVM is a type of SVM that can handle non-linearly separable data by mapping it to a higher-dimensional space, where it may become linearly separable.

The basic idea behind SVM is to find a hyperplane that separates the data points of different classes with the maximum margin. In a non-linear SVM, the data is first transformed into a higher-dimensional space, where it can be linearly separated. This transformation is usually achieved by using a kernel function that maps the data to the higher-dimensional space.

The most commonly used kernel functions in non-linear SVM are the radial basis function (RBF) kernel and the polynomial kernel. The RBF kernel maps the data to an infinite-dimensional space, while the polynomial kernel maps it to a higher-dimensional space with a fixed degree.

Once the data is mapped to the higher-dimensional space, the SVM algorithm tries to find a hyperplane that separates the data points of different classes with the maximum margin. This hyperplane is defined by a set of support vectors, which are the data points closest to the hyperplane.

Non-linear SVM can provide a more flexible and accurate model than linear SVM, especially when dealing with complex and non-linear data. However, it may be more computationally expensive and prone to overfitting, which means that it may perform well on the training data but not on new unseen data. Therefore, it is important to carefully select the kernel function and other hyperparameters to optimize the performance of a non-linear SVM model.

Linear SVM

Using an example, you may better understand the SVM algorithm. Imagine a dataset with two distinct colours and two tags (green and blue) (x_1 and x_2). It is necessary to provide a classifier that can tell apart between coordinate pairs in blue and green. (x_1, x_2). It can be simply distinguished between these two classes using a straight line as it is a 2-d space. To divide these classes, however, there may be more than one line. The most suitable perimeter or region is thus identified using the SVM approach, and this best boundary or area is known to be a hyperplane. Areas where both groups' lines are most similar to one another is determined by the SVM algorithm. They are referred to as "support vectors." The edge is the distance between the vectors and the hyperplane. SVM aims to increase this margin. The best hyperplane is the one with the highest margin. Drawing a straight line can be used to separate data that is organised linearly, however non-linear data cannot be divided by a straight line. Thus, to distinguish between these information points, it must provide a further dimension. It utilised the two dimensions x and y for data that

is linear, and will include the third dimension, z , for non-linear data. It is established that:

$$z=x^2+y^2 \quad (2)$$

The sample space will resemble this by including a third dimension. SVM is now going to divide the datasets into a number of classes. Since they are in three-dimensional space, it seems to be a plane parallel to the x -axis. It will change to the following if they are converted it to 2D space with $z=1$. Finally, it can determined the location of the eyes using this approach.

By using this algorithm, it will get the landmarks of the face very easily that too readily available, it just have to perform some functions to automate the mouse events.

4.2 Deep Neural Network

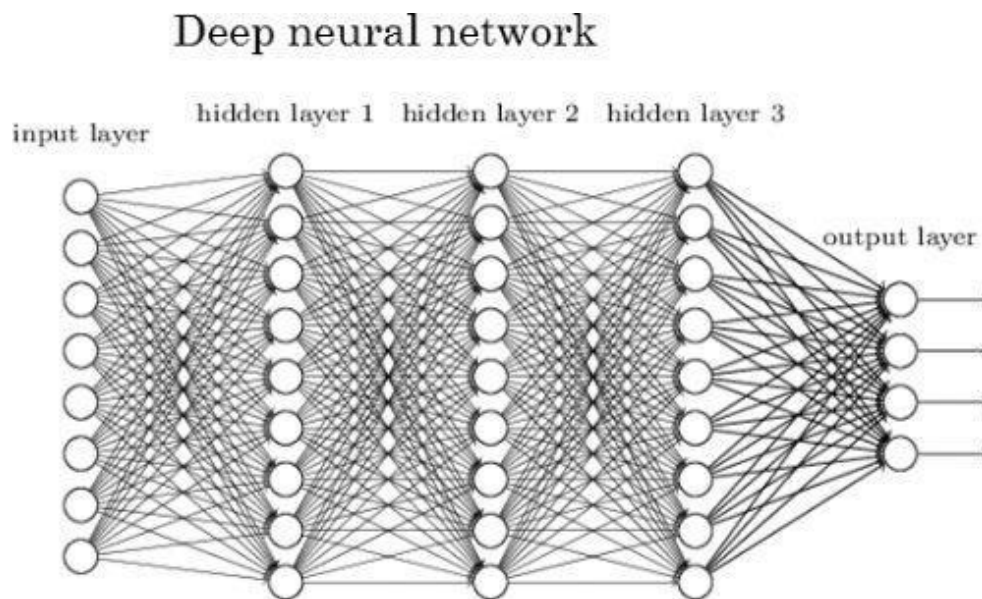


Fig.4.1 Deep neural network

Nodes are little parts of the system, and they are like neurons in the human brain. When a stimulus hits them, a process takes place in these nodes. Some of them are connected and marked, and some are not, but in general, nodes are grouped into layers.

The system must process layers of data between the input and output to solve a task. The more layers it has to process to get the result, the deeper the network is considered. There is a concept of credit assignment path (CAP), which means the number of such layers needed for the system to complete the task. The neural network is deep if the CAP index is more than two.

A deep neural network is beneficial when you need to replace human labour with autonomous work without compromising its efficiency. The use of deep neural networks can find various applications in real life. For example, the Chinese company Sense Time created a system of automatic face recognition to identify criminals that uses real-time cameras to find an offender in a crowd. Nowadays, it has become a popular practise in the police and other governmental entities. The American company [Pony.ai](#) is another example of how you can use DNN. They developed a system for AI cars that can work without a driver. It requires more than

just a simple algorithm of actions, but a much deeper learning system, which should be able to recognize people, road signs and other markings like trees, and other important objects.

The famous company UbiTech creates AI robots. One of their creations is the Alpha 2 robot, which can live in a family, speak with its members, search for information, write messages, and execute voice commands.

4.3 Deep Neural Network methodology

The neural network needs to learn all the time to solve tasks in a more qualified manner or even to use various methods to provide a better result. When it gets new information in the system, it learns how to act accordingly in a new situation.

Learning becomes deeper when you solve harder tasks. A deep neural network represents the type of machine learning where the system uses many layers of nodes to derive high-level functions from input information. It means transforming the data into a more creative and abstract component.

In order to understand the result of deep learning better, let's imagine a picture of an average man. Although you have never seen this picture and his face and body before, you will always identify that it is a human and differentiate it from other creatures. This is an example of how the deep neural network works. Creative and analytical components of information are analysed and grouped to ensure that the object is identified correctly. These components are not brought directly to the system; thus, the ML system has to modify and derive them.

4.4 Linear Discriminator

Linear discriminant analysis (LDA) is a dimensionality reduction technique. As the name implies, dimensionality reduction techniques reduce the number of dimensions (i.e., variables) in a dataset while retaining as much information as possible. For instance, suppose that we plotted the relationship between two variables where each colour represents a different class.

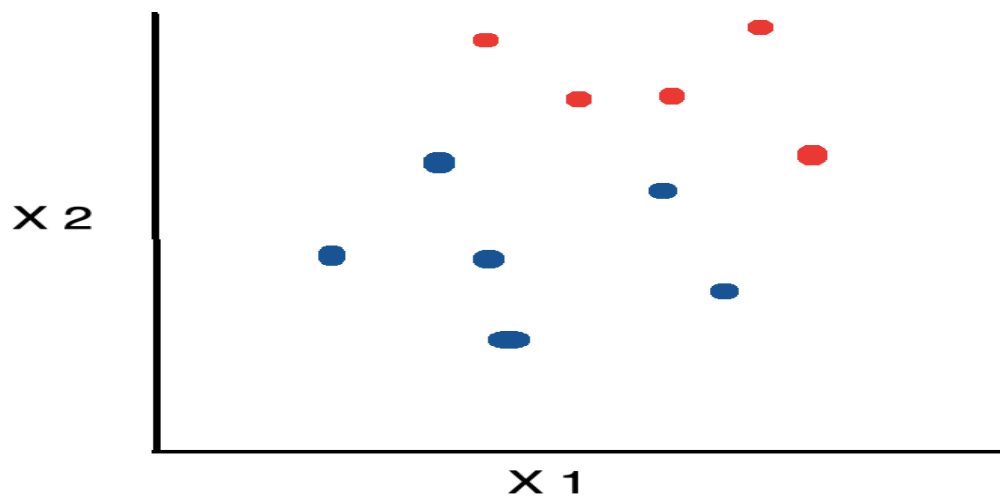


Fig.4.2 Graph of dataset

If we'd like to reduce the number of dimensions down to 1, one approach would be to project everything on to the x-axis.

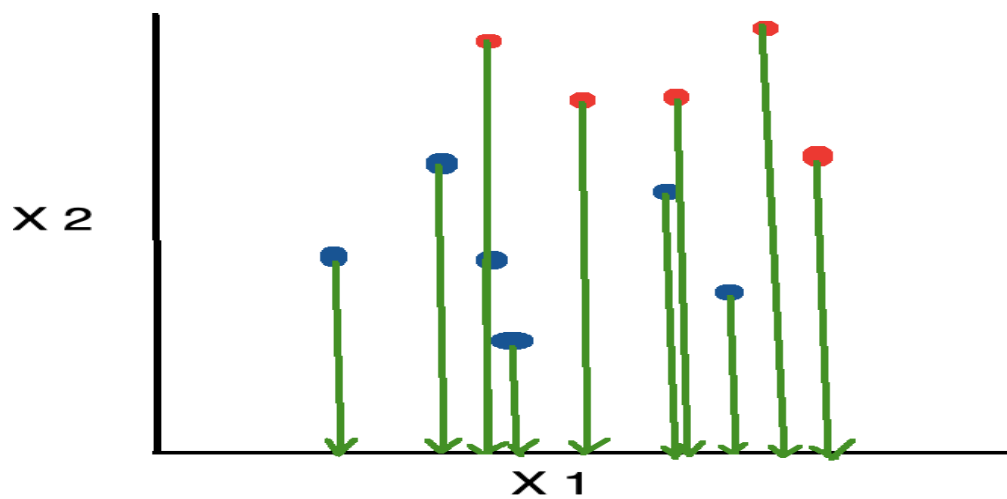


Fig.4.3 LDA: Projection on X-Axis



Fig.4.4 Dimensionality Reduction

This is bad because it disregards any useful information provided by the second feature. On the other hand, Linear Discriminant Analysis, or LDA, uses the information from both features to create a new axis and projects the data on to the new axis in such a way as to minimize the variance and maximize the distance between the means of the two classes.

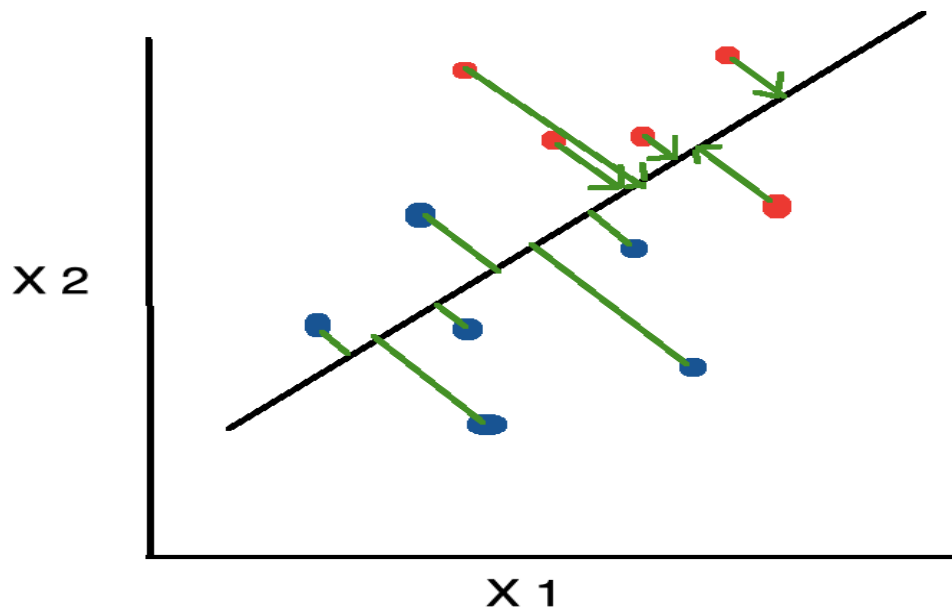


Fig.4.5 Linear Discriminant Analysis in Python

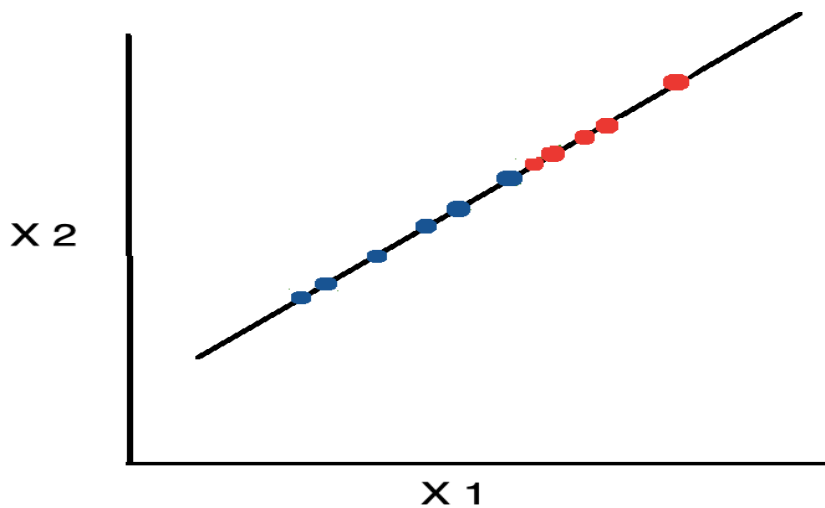


Fig.4.6 Minimizing variance



Fig.4.7 LDA: Better Output

By doing this Linear Discriminant Analysis, it can be estimated the probability that a new set of inputs belongs to every class. So, after doing every step, the blue dots and red dots based on their classes can be separated. After separating, it can be recognised that the object after the classification based upon classes using LDA which uses Bayes' Theorem to estimate the probabilities.

BLOCK DIAGRAM

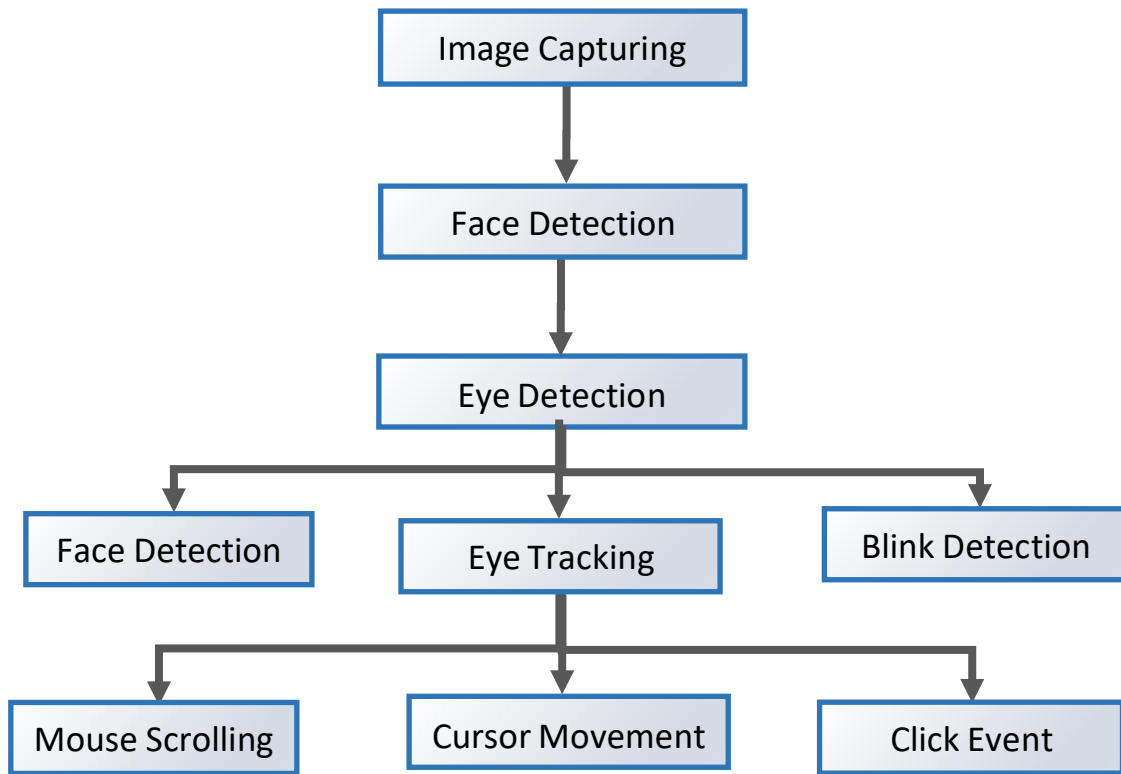


Fig.4.8 Block diagram of the proposed system

The user has to sit in front of the display screen of a private computer with a specialised video camera established above the screen to study the consumer's eyes. The laptop constantly analyses the video photo of the attention and determines where the consumer is calling on the display screen. Nothing is attached to the consumer's head or body. To Frames. To "pick out" any key, the user looks at it for a specific amount of time, and to "press" any key, the consumer simply blinks the eye. On this device, calibration procedures are not required. For this system, the entry is the easiest. No Outside Hardware Is Connected or Required. The camera gets its input from the eye. After receiving these streaming movies from the cameras, it'll spoil Into Frames. After Receiving Frames, It Will Check for Lighting Conditions Because Cameras Require Enough Lighting Fixtures from External Sources, in Any Other Case, Blunders Will Show on the Screen. The captured frames, which can already be in RGB mode, are transformed into black and white.

Five pictures (frames) from the Enter Supply focusing the eye are analysed for iris detection (middle of the eye). Following that, a mid-point is calculated by taking the suggested left and right eye centre points. Eventually, the mouse will move from one position on the display to another, and the customer will click by blinking their eyes for 5 seconds.

CHAPTER 5

SOFTWARES AND ALGORITHMS USED

SOFTWARE REQUIREMENTS

- Python
- OpenCV
- PyAutoGUI

Python

Python is a high-level, interpreted programming language known for its simplicity, readability, and versatility. It was first created in the late 1980s by Guido van Rossum and has since become one of the most popular languages for a wide range of applications, including web development, data analysis, machine learning, scientific computing, and more.

One of the key features of Python is its simplicity and ease of use. Its syntax is easy to learn and read, making it an ideal language for beginners. The language emphasizes code readability, which means that it is designed to be easily understood and maintained by developers.

Python is also known for its vast collection of libraries and frameworks, which provide a wealth of functionality for developers. Some of the most popular libraries include NumPy, Pandas, Matplotlib, and SciPy, which provide powerful tools for data analysis and scientific computing. Additionally, Python has a large and active community of developers, who contribute to the language by creating new libraries and tools, as well as supporting each other through online forums and communities.

Another advantage of Python is its cross-platform compatibility. Python code can run on various operating systems, including Windows, Linux, and macOS, without the need for modification. This makes it a popular choice for developing software that needs to run on multiple platforms.

Finally, Python is an interpreted language, which means that it does not need to be compiled before running. This allows for rapid prototyping and iteration, making it an ideal choice for developing software in an agile environment.

Python OpenCV

OpenCV (Open Source Computer Vision Library) is a popular open-source computer vision and machine learning software library, originally developed by Intel in 1999, and now maintained by the OpenCV community. It is written in C++, but also provides bindings for various programming languages, including Python.

Python OpenCV is a Python interface for the OpenCV library that allows developers to use OpenCV functions and algorithms in Python code. It provides a rich set of image processing and computer vision algorithms that can be used for various tasks such as object detection, recognition, tracking, face detection, image and video analysis, and more.

Python OpenCV provides various modules for image and video processing, including image filtering, feature detection and description, object recognition, motion analysis, and more. It also provides support for reading and writing various image and video file formats, as well as for capturing live video streams from cameras.

One of the main benefits of using Python OpenCV is its simplicity and ease of use. It provides a Pythonic interface that makes it easy to write Python code for image processing and computer vision tasks. Additionally, it has a large community of developers who contribute to its development, provide support, and share their knowledge and code through various forums and resources.

In summary, Python OpenCV is a powerful tool for image and video processing and computer vision tasks that provides a simple and easy-to-use Python interface for the OpenCV library.

PyAutoGUI

PyAutoGUI is a Python library that allows you to automate mouse and keyboard operations on a computer. It provides a simple and cross-platform interface for controlling GUI applications, automating repetitive tasks, and testing software. Some of the common tasks that can be automated using PyAutoGUI include clicking, typing, dragging, scrolling, and taking screenshots.

PyAutoGUI works by using Python's built-in modules such as `pywin32` (for Windows) and `Quartz` (for Mac) to simulate keyboard and mouse input events. It can also retrieve information about the screen, such as the position of windows, mouse cursor position, and colour values of pixels. This allows for a wide range of possibilities when automating tasks.

One of the main advantages of PyAutoGUI is its simplicity. Its easy-to-use functions and intuitive interface make it accessible to beginners who have little or no experience in automation. Additionally, PyAutoGUI is cross-platform, meaning that it can be used on Windows, Mac, and Linux operating systems.

However, it is important to note that PyAutoGUI may not work well in some situations, especially in cases where there is a delay or lag in the system. It is also important to use PyAutoGUI carefully, as it can potentially cause damage to the system if not used properly.

PyAutoGUI Library

The PyAutoGUI library in Python is an automation library that supports mouse and keyboard control. Alternatively, it enables us to automate the movement of the mouse and keyboard in order to establish interaction with the other application using the Python script. It has many features, some of which are listed below.

- We can move the mouse and click in the windows of the other applications.
- The keystrokes can be routed to other applications. For example, filling out a form, entering a search query into a browser, and so on.
- We can also take photographs and provide an image.

- It allows us to locate an application window and move, maximise, minimise, resize, or close it.
- Show the alert and message boxes.

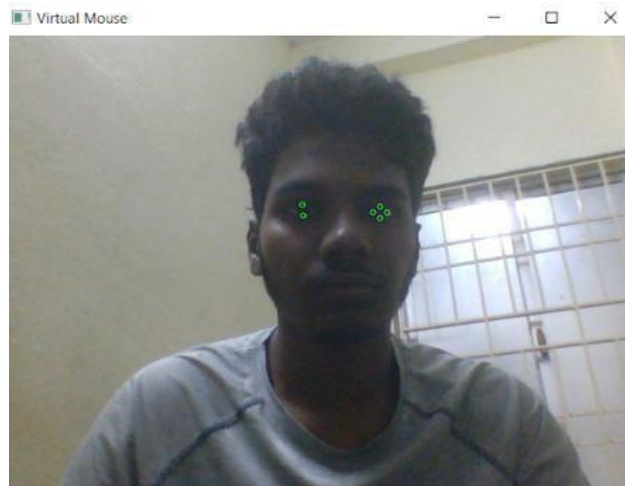
Algorithms

- We used SVM algorithm.
- Dataset esteems changed over into exhibit esteems which is going to give to the calculation to discover precision.
- Select the calculation dependent on the exactness and dissect the information by utilizing the calculation.

CHAPTER 6

RESULTS AND CONCLUSION

In this research, First, it will locate the eye's pupil in the middle. Then, various commands are established for the virtual keyboard depending on the variations in pupil position. The signals pass through the motor driver on their way to the virtual keyboard itself. The motor driver's ability to regulate direction and speed allows the virtual mouse to go forward, left, right, and stop. It provides a control method based on eye tracking that enables people to easily and intuitively interact with computers using only their eyes. The system combines keyboard and mouse functionality such that users may use it to do practically all computer inputs without the need for conventional input devices. The technology not only makes it possible for disabled people to use computers in the same way as able-bodied users, but it also gives sighted persons a fresh option for using computers. In a surfing trial, the suggested method increased browsing effectiveness and enjoyment, and also allowed users to easily engage with multimedia.



Finally, it can determined the location of the eyes and do mouse events using this approach.

REFERENCES

- [1] Brooks,R.E.(1997) —Towards a theory of the cognitive processes in computer programming,|| Int. J. Man-Mach. Studies, vol. 9, pp. 737–751.
- [2] Cheng- Chih Wu, Ting-Yun Hou(2015)||Tracking Students' Cognitive Processes During Program Debugging||—An Eye-Movement Approach, IEEE.
- [3] D. LeBlanc, H. Hamam, and Y. Bouslimani, "Infrared-based humanmachine interaction," in Proc. 2nd Int. Conf. Inf. Commun. Technol., 2006, pp. 870–875.
- [4] Ehrlich,K. and Soloway,E.(1983) Cognitive strategies and looping constructs: An empirical study,|| Commun. ACM, vol. 26, no. 11, pp. 853–860.
- [5] Eric Sung and Jian-Gang Wang (2002)—Study on Eye Gaze Estimation||, IEEE, VOL. 32, NO. 3, JUNE .
- [6] Murphy,L. (2008)—Debugging: The good, the bad, and the quirky—A qualitative analysis of novices' strategies,|| SIGCSE Bull., vol. 40, no. 1, pp. 163–167.
- [7] N. Shaker and M. A. Zliekha, "Real-time Finger Tracking for Interaction," 2007 5th International Symposium on Image and Signal Processing and Analysis, Istanbul, 2007, pp. 141-145.
- [8] QiangJi and Zhiwei Zhu (2007)||Novel Eye Gaze Tracking Techniques Under Natural Head Movement||, Senior Member, IEEE, VOL. 54, NO. 12, DECEMBER
- [9] R. S. Batu, B. Yeilkaya, M. Unay and A. Akan, "Virtual Mouse Control by Webcam for the Disabled," 2018 Medical Technologies National Congress (TIPTEKNO), Magusa, 2018, pp. 1-4.

- [10] S. Yousefi, F. A. Kondori, and H. Li, "Camera-based gesture tracking for 3D interaction behind mobile devices," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 26, no. 8, Dec. 2012, Art. no. 1260008.
- [11] S. K. Kang, M. Y. Nam and P. K. Rhee, "Color Based Hand and Finger Detection Technology for User Interaction," 2008 International Conference on Convergence and Hybrid Information Technology, Daejeon, 2008, pp. 229-236.
- [12] Z. Ma and E. Wu, "Real-time and robust hand tracking with a single depth camera," *Vis. Comput.*, vol. 30, no. 10, pp. 1133–1144, Oct. 2014.