

A picture containing text

Description automatically generated

[ **Idea 3: EOG based interface ]**

Icon

Description automatically generated with medium confidence

***Scientific Computing Department***

***Faculty Of Computer and Information Science***

***Ain Shams University***

***Team* (32) SC**

***HCI Project***

***Under Supervision:***

***Dr Manal Tantawi***

***TA Manar Sultan && TA Aya Naser***

*From* ***OT3A*** *Team*

|  |  |  |
| --- | --- | --- |
| Name | ID | Section |
| Osama Anter Mohamed Afify | 20191700091 | 1 |
| Tarek Ashraf Mahmoud Hussein | 20191700322 | 3 |
| Ahmed Mohamed Ibrahim Mohamed | 20191700059 | 1 |
| Adham Mohamed Tawfik Mohamed | 20191700086 | 1 |
| Ahmed Mohamed Ali Abdelrahman | 20191700068 | 1 |

**Idea of Project:**

If we have an **EOG signal**, it is a signal of eye movements up, down, right, left, or blink and we want to use it to set up specific tasks so that The UI should enable the user to choose whether to sleep, eat, drink water, or go to the bathroom by UI Design.

**Dataset:**

We have 250 file Each Signal have 5 Classes:

Yukari --> Up

Asagi --> Down

Sag --> Right

Sol -->Left

Kirp -->Blink

**1-Data preparation and Preprocessing:**

Data Preparation:

* The dataset is imported using the panda’s library and read\_csv function.
* The dataset is split into training and testing sets using the train\_test\_split function from the sklearn library.
* The training data is used to fit the model and the testing data is used to evaluate the model's performance.

Preprocessing:

* The dataset is checked for null values and missing data using the isnull () function.
* Any missing data is handled using the fillna () function from the pandas library.
* The dataset is scaled using the StandardScaler function from the sklearn library which standardizes the features by removing the mean and scaling to unit variance.

We Use This Techniques to Clean Data and Preprocessing Text:

* Remove Rows Have Nulls Cells.
* Removing Duplicate Rows.
* Label Encoder.

**2- Feature extraction methods:**

1. The feature extraction method used is Principal Component Analysis (PCA) which is a dimensionality reduction technique.
2. The number of principal components used is determined by setting the n\_components parameter of the PCA function.

**3- Classification Models:**

* **KNN:**
* **Support Vector Machine Classifier** **(SVM) algorithm:**
  + The parameters used for the SVC classifier are: kernel='rbf', C=1.0, and gamma='scale'.

SVM is a supervised machine learning algorithm that can be used for both classification and regression tasks. SVM works by finding the best boundary or hyperplane that separates the data points of different classes with the maximum margin. The margin is the distance between the hyperplane and the closest points from each class.

**A picture containing screenshot, diagram, visualization

Description automatically generated**

* **Random Force:**

**4- Classification results:**

1. **KNN**
2. **Support Vector Machine Classifier Classification Results:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Precision | Recall | F1-Score | Support |
| 0 | 0.70 | 0.91 | 0.79 | 178 |
| 1 | 0.75 | 0.41 | 0.53 | 96 |
| Accuracy |  |  | 0.71 | 274 |
| Macro Avg | 0.72 | 0.66 | 0.66 | 274 |
| Weighted Avg | 0.71 | 0.71 | 0.68 | 274 |

1. **Random Force**

**5- UI:**

**Thank You :)**