

MORSE CODE BASED SECURED AUTHENTICATION SYSTEM THOUGH ARTIFICIAL INTELLIGENCE

Project Reference No.: 45S_BE_2679

College : *Rajiv Gandhi Institute of Technology, Bengaluru*
Branch : *Department of Computer Science and Engineering*
Guide(s) : *Dr. D G Anand*
Student(S) : *Mr. T N Nivas Darshan*
Mr. Madhu Kiran B
Mr. Manjunath S

Introduction:

Data science is a multidisciplinary blend of data inference, algorithm development and technology in order to solve analytically complex problems and also Face recognition has been one of the most interesting and important research fields in the past two decades. In this project, we'll go through general ideas and structures of recognition, important issues and factors of human faces, critical techniques and algorithms, where the individual have to get registered their details and later during login if the information gets satisfied with the previously registered details it moves on to the next step of Morse code. If the information does not match with the registered details it captures the picture and is sent to the to the authorized person through email and finally gives a comparison and conclusion. Gaze- based authentication refers to finding the eye location across sequential image frames, and tracking eye center over time. Password authentication will be done using Morse code, where numbers will be represented in dots and dashes. This model presents a real-time application for gaze-based PIN entry, and eye detection and tracking for PIN identification using a smart camera. Since most of the people in the world are facing problems in the field of authentication and security. We are able to provide a real time eye tracing for password authentication for people who authenticate themselves using Morse code.

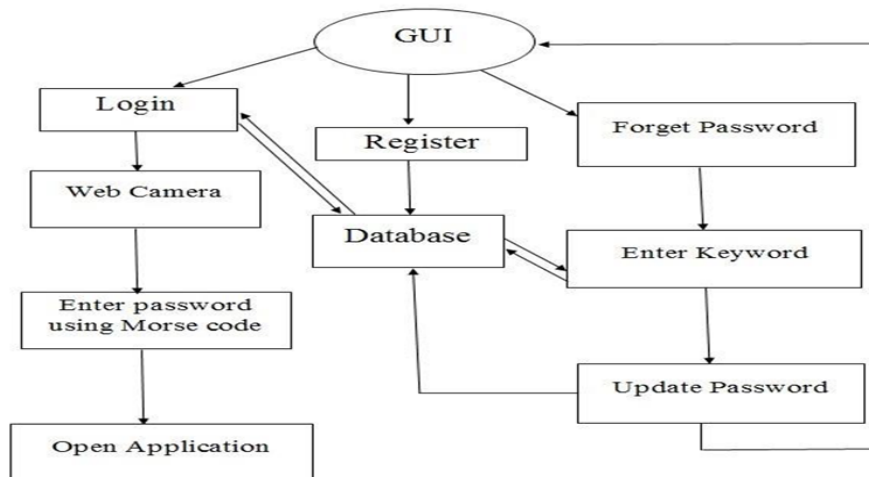
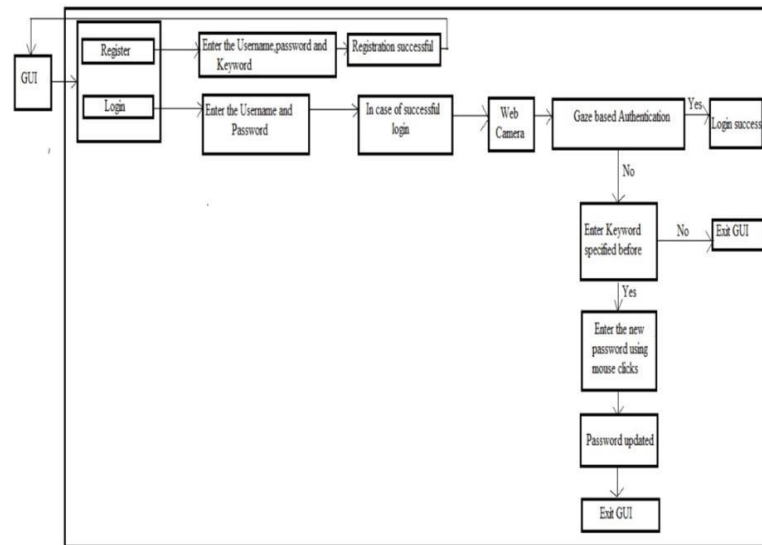
Objectives:

1. To create a secure system to authenticate users who are not completely blind.
2. To create a secure password authentication system which uses Morse code.
3. To make sure that the required parts of the face are recognized accurately by the system.

Methodology:

1. It represents the architecture or the basic design that is required for the implementation of the model. The model consists of a user interface and back end database.
2. GUI is created such that the user can interact with the system. Pygame or OpenCV is used in to create it.

- In the frontend firstly the user need to register by providing a user id of choice, a password (PIN) and a keyword. After registration the user can log in by using the credentials i.e. user id and password. With the help of a web camera the PIN is taken as input in the form of Morse code.
- In the backend, the entered PIN is checked with the stored PIN which was entered into the database by the user while registering. If the entered PIN is not correct, its exits the screen. If the entered PIN is correct, it displays successful authentication. If the user has forgotten his password then he can use the keyword to authenticate and update the existing password with a new one.



Conclusion:

Our project basically provides two factor authentication. Two factor authentication is basically providing, Project is also helpful for physically disabled people in order to authenticate. People can make use of this model who have basic knowledge on morse code. Concerning the future enhancement, we are trying to implement facial recognition for each user, there will be no need to enter the password at all. We are also trying to deploy this model in government sectors, with a smaller number of steps required for authentication.

```

C:\Windows\System32\cmd.exe - python Morse_login.py
p.7075702075702076
p.7075702075702076
p.695006080813077
p.695006080813077
text1 ['- ', '- ', '- ', '- ', '- ']
p.9056490384615384
p.9056490384615384
l.059047619047619
l.059047619047619
l.059047619047619
p.0092105263157895
p.0092105263157895
l.0289968652037618
l.0289968652037618
l.3136363636363635
l.3136363636363635
p.0438405797101449
p.0438405797101449
p.0438405797101449
l.2535211267605635
text1 ['- ', '- ', '- ', '- ', '- ']
selection of Single no is completd
<class 'str'>
selected no ['- ', '- ', '- ', '- ', '- ']
password ['0', '0']
got the password and i 00,00
type the xhar password and i <class 'str'> ,<class 'str'>
password matches
[ WARN:1056.881] global D:\a\opencv-python\opencv-python\opencv\modules\videoio\src\cap_msmf.cpp (539) `anonymous-namespace`::SourceReaderCB::~SourceReaderCB terminating async callback

```

```

C:\Windows\System32\cmd.exe - python Morse_login.py
Microsoft Windows [Version 10.0.19043.1706]
(c) Microsoft Corporation. All rights reserved.

E:\Morse_with_face_recognition_updated>python Morse_login.py
01.79390725030164
unknown
mail Sent
00.94030545046561
unknown
mail Sent
09.121143886636595
unknown
mail Sent
09.121143886636595
unknown
mail Sent
04.542605631099836
Success
Login Successful!!!!
[ WARN:1056.571] global D:\a\opencv-python\opencv-python\opencv\modules\videoio\src\cap_msmf.cpp (539) `anonymous-namespace`::SourceReaderCB::~SourceReaderCB terminating async callback
check ['manja', '00'],type <class 'list'>
password Not Recognised.
Enter the Security Answer
Pet name or nick name
na

```

```

Instructions:
Dot (.)           : Left Click
Dash (-)         : Double Left Click
Next Letter      : Right Click
Next Word        : Double Right Click

```

Scope for future work:

This project is also helpful for disabled people in order to authenticate. People from kids to old people can make use of this model who have basic knowledge on morse code. For blind people, there are keyboards with braille dots present on each button.

Concerning the future enhancement, we are trying to implement facial recognition for each user, there will be no need to enter the password at all.

We are also trying to deploy this model in government sectors, with a smaller number of steps required for authentication.