Wireless Online Shopping for Visually Impaired People Using Natural Language Processing and Face Recognition Mechanism

1st Kirat Jadhav
dept. Electrical and Computer Engineering
Stevens Institute of Technology
Hoboken, USA
kjadhav1@stevens.edu

2nd Kevin Lu
dept. Electrical and Computer Engineering
Stevens Institute of Technology
Hoboken, USA
klu2@stevens.edu

***Abstract*—The following project aims to create a robust optimal model for making online shopping easier for visually impaired people. This paper provides a detailed comparison of the algorithms and libraries used for speech recognition, text-to-speech, browser automation, and face recognition. Determination of the best speech recognition and text-to-speech libraries are done using the concept of word error rate, which is the ratio of mistakes in a transcript to the total number of words uttered. As the purpose of this paper is to make the lives of visually impaired people easier, we use different face recognition mechanism to try and implement the one that suits our unique specifications. It includes a comprehensive research for determining the most effective possible facial recognition mechanism for visually impaired people while taking into consideration numerous challenges that they face in real-world settings. Later, we intend to automate the whole online shopping process using different web scarping and web driver tools.**

***Index Terms*—NLP, face recognition, LBPH, word error rate, web scrapping, selenium, gTTS**

I. Introduction

Over the past decade, online shopping has become more and more common. The heavy use of a keyboard and mouse from login through product selection prevents visually challenged people from using it. This project attempts to create solutions that improve these e-commerce websites' usability for those who are visually impaired. The paper compares various face recognition algorithms and text-to-speech libraries in order to filter and get an optimal outcome that will be more accurate. We use the word error rate formula, a common metric we applied to the unique dataset we developed, to assess the precision of a voice recognition library. Typically, there are several steps involved in utilizing an e-commerce website, starting with logging in and continuing with product selection and rating filtering. The project entails on creating a fully voice-based implementation of the various e-commerce website phases. The website will be fully automated, free from the usage of a keyboard or mouse, and able to web scrape information from an e-commerce website.

A. Related Work

There were contributions made earlier in the field of face recognition and natural language processing. In this project, we have combined both the techniques to facilitate user friendly interaction between the user and the computer. Precious related work include Howse used OpenCV library along with Arduino and Haar filter. The whole project was programmed on Arduino [2]. Petrovic and Stanisevic provided method to scrap data from various parts from the internet and store it in the database. This was used to analyse data from car market. Sarkar proposed implementation of reading device for the visually impaired using Google Translates text-to-speech module [3]. Ali and El-Hafeez proposed a system for pattern detection as well as face and eyes detection using Haar cascade [5]. Ramya and Sindhura proposed a software development cycle using Selenium web drivers for testing [6]. K. Lee and C. Lee offered a fast object detection algorithm based on local binary patterns and color histograms [7]. Belhumeur and Hespanha presented a comparative analysis of Eigenfaces and Fisherfaces face recognition mechanisms [8]. Dordinejad and Cevikalp implemented a image based face recognition using three dimensional and generative adversarial network [9]. Naik and Guinde explored various LBPH alogirthm variation from side to frontal features using a GPU [10].

B. Organization

Generally, using an e-commerce website involves multiple stages from log in to selection of the product. The project entails towards a fully voice-based implementation of the various e-commerce website phases. The rest of the project is followed by the problem description and the preliminary ideas and results including the algorithms, text-to-speech and speech to text selection, face recognition model mechanism, browser automation and Implementation results.

II. Problem Description

One of the communities that is heavily discriminated from accessing websites are visually impaired people. According to data published in National library of Medicine, around

253 million people are suffering from visual impairment in the whole world. Out of which 36 million are completely blind, while the rest 217 million have moderate to severe visual impairment (MSVI) [1]. According to a 2016 National Institute of Blind report, there are 7,675,600 cases of people with visual disabilities in the United States of America itself. Since most of the websites today are not visually impaired friendly, lots of these people need constant assistance for accessing any website. This is the main reason people with visual disability do not prefer to shop online. The United States government initiated a project where they were set to make 100+ government websites user-friendly for the visually impaired people. Today, hardly any government websites are user-friendly to these people.

It is now critical to make these websites accessible to everyone in the United States and around the world, regardless of ability. In this project, we present a robust optimal model that will be implemented as functional software for a wireless online purchasing system. We include a Local Binary Patterns Histograms (LBPH) based facial recognition algorithm into the OpenCV module for the graphical user interface, that will assist in identifying people in real time using the device's web camera. For navigation in the Chrome browser, we utilize Selenium web drivers, that help to totally automate the procedure without the use of a keyboard or mouse. For communication between the user and the software, we additionally use gTTS and text-to-speech libraries. Various voice recognition libraries are reviewed and compared in order to discover the best one with the lowest cost and highest accuracy. To verify the accuracy of text-to-speech libraries, we implement on our own voice data set.

III. Preliminary Ideas and Results

A. Algorithm

The software automation algorithm is depicted in Fig. 1. The method begins with a facial recognition login or registration step. Then, using text-to-speech, ask the user what he wants to search for. He will be prompted if he wants to look for a certain product. The customer will be asked to identify any specific brand for the product once he gives the product name. If he answers no, the program will conduct a random search and present the relevant results. If he answers yes, the show will feature that specific product with the specified brand. Later, the user will be asked whether he wants to specify a budget range for the product he is looking for. If no, the product is presented at random, but if yes, the filters are set accordingly. When this procedure is finished, the merchandise will be automatically loaded to the cart. This entire automation is carried out using the Google Chrome browser, that itself is available on all devices and interfaces fairly well with the Selenium web drivers, that assisted us in scraping the data from the e-commerce website here.

B. Speech Recognition and Text-to-Speech

Speech Recognition plays a crucial role in this project since it serves as a communication medium between the user and the system. The best speech recognition and text-to-speech libraries are always determined based on a single factor that is, accuracy. Many factors influence speech accuracy, such as audio quality of the input, unwanted background noise, and the diverse accents in the same language for example in the language English there are many accents like American, British, Indian, etc. Speech recognition systems are sensitive to input. So sometimes few speech recognition gives different accuracy for different accents. There are many different speech to text and text-to-speech libraries out there, many of them are paid and many are free too. The paid libraries generally does provide better accuracy because they have a big data to train there model. Cost of speech recognition is a factor here we consider as we intend to make a cost effective solution. The few examples of speech libraries are as follows:

- IMB Watson's Speech to Text API
- Google Cloud's Speech to Text API
- Google Translate's text-to-speech (gTTS)
- SpeechRecognition library
- Pyttsx3 library

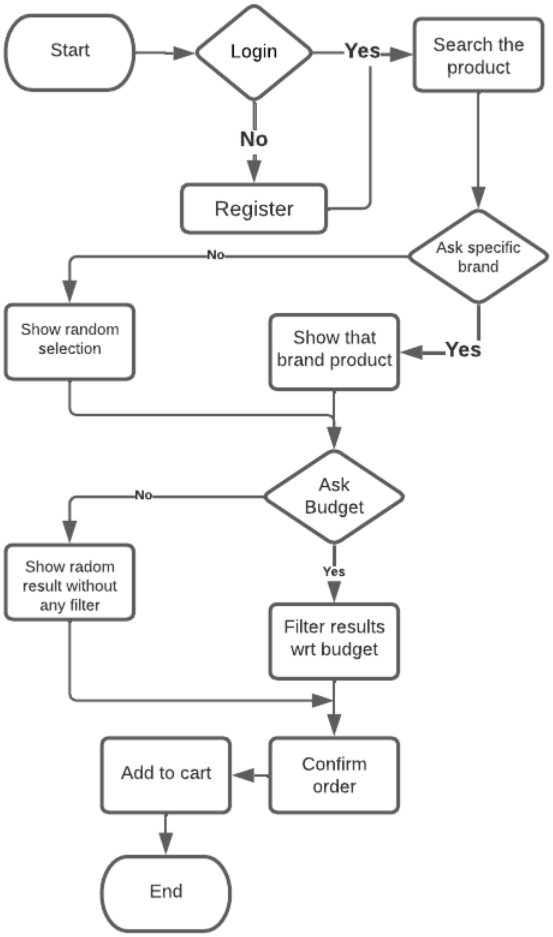


Fig. 1. Algorithm flowchart.

To evaluate the accuracy, we should have a common language to compare the quality of the speech. Therefore, use the concept of word rate error (WER). WER is a mathematical metric that is used to find out the performance of speech library, basically it notes the error in the transcript and compares it to the speech input. The WER is the mean of total error divided by the total number of words in the data (1). The S is the number of substitutions that is the number of words that are present but missed. The D is number of words that is missing in the hypothesis. The I is the number of extra words that has been inserted. The N is the total number of words in the data.

$$Word\ Error\ Rate\ = \frac{S+D+I}{N} \tag{1}$$

where,

S = number of substitutions
D = number of deletions
I = number of insertions
N = total number of words

We created our own data set that includes speech of three different accents to test this word error rate formula. We use the ’jiwer’ Python library to calculate WER. We test and implement these on all of the above mentioned speech libraries on our data set. WER should be as low as possible and WER between 5% and 15% is seen as a good quality speech and ready to use, while anything above 20% is seen as a poor quality. WER can be greater than 100% in a situation of very poor audio quality. The rest of the% after the WER is the accuracy. Fig. 2, shows the comparisons of all the libraries.

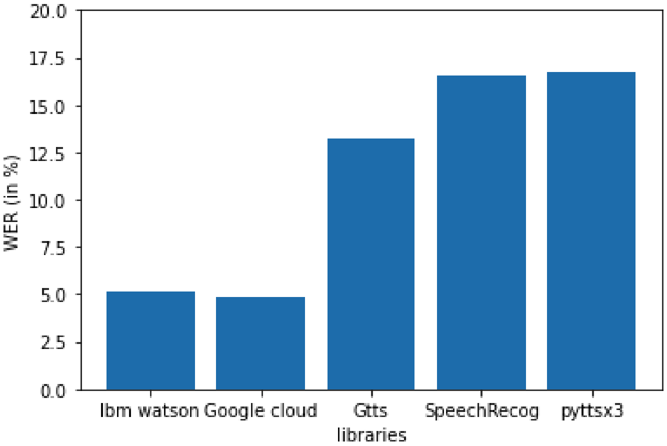


Fig. 2. Comparisons of word error rates.

From the Fig 2, we can evaluate that Google Cloud’s Speech to Text is best with the lowest WER of 4.9%, while the IBM Watson’s Speech to Text is close with 5.1% WER. These two particular APIs use very big data set to train there that helps in gaining minimum error. The gTTS receives a WER of 13.2% on our model, whereas the SpeechRecogniton library and the pyttsx3 library got WER of 16.5% and 16.7% respectively. Among these the Google Cloud’s Speech to Text, and IBM Watson’s Speech to Text are paid after certain amount of characters so we can’t choose them based on the cost factor. The gTTS and the SpeechRecogniton library gives us the least WER as well as they are free to use libraries. That’s why we select Google Translate’s text-to-speech and SpeechRecogniton library in this project.

Using SpeechRecognition library, the user can now talk with the e-commerce website. Speech will be recognized and translated to text that will help in searching of the product. Following the search, the user hears whatever text output was obtained by utilizing the gTTS library to translate it to speech. The gTTS library converts the text from the input to speech and return it in the form of string in Python.

There are many advantages of using Google Translate’s text-to-speech library as it is developed by Google and our browser in this particular project is Google Chrome, helps in smooth integration into each other. The disadvantage of it is it requires constant internet to work but it doesn’t affect the project as this is project will always require internet to navigate through the e-commerce website.

C. *Face Recognition Mechanism*

Face recognition is one of the leading technologies. There used everywhere in the modern day and age. We use Open Source Computer Vision Library, i.e., OpenCV library. OpenCV is enormously used in the field of machine learning and computer vision. There are two main parts for face recognition, the first is the face detection whose primary role is to face a human face in a picture or camera feed. We use Haar cascade classifier, that helps to train our model by feeding various images with face. OpenCV contains its very own trainer and detector where we integrate it with Haar cascade classifier. Once the image is loaded first it converts the image to grayscale format and goes through every pixel to detect the face. It has an accuracy of 95% for large amount of data and with a false positive of 1% [4]. The next part of the mechanism is the recognition portion. Over the year, there have been many different techniques to implement face detection, e.g., Eigenfaces, Local Binary Pattern Histograms, Fisherfaces, etc.

Linear Binary Pattern Histograms (LBPH) when first introduced was used for similar pattern recognition in 1994 [5]. Later, it was integrated with the histogram of oriented gradient descriptor. LBPH first starts with highlighting the characteristics of the input and then later dividing it into binary form. The input image is divided four parts, i.e., Radius, Neighbor, GridX, and GridY. The radius is responsible for the layout of the pattern, that generally starts from the center. The Neighbor are the number of data points located inside the radius formed of the local binary pattern. GridX and GridY divide the image into horizontal and vertical plane. Eigenfaces and Fisherfaces algorithm are few of the other options for face recognition.

Table I compares accuracies of LBPH mechanism, Eigenfaces mechanism, and Fisherfaces mechanism on datasets with different volumes of data. Dataset 1 demonstrates that LBPH

TABLE I
DATASET RESULTS

Dataset 1	***LBPH***	**Eigenfaces**	**Fisherfaces**
Total Images	25	25	25
Recognized	24	23	21
Error	1	2	4
Accuracy	96%	92%	84%
Dataset 2	***LBPH***	**Eigenfaces**	**Fisherfaces**
Total Images	60	60	60
Recognized	59	57	54
Error	1	3	6
Accuracy	98.33%	95%	90%
Dataset 3	***LBPH***	**Eigenfaces**	**Fisherfaces**
Total Images	150	150	150
Recognized	149	148	144
Error	1	2	6
Accuracy	99.33%	98.66%	96%

correctly identified 24 out of a possible 25 images, with a 96% accuracy rate. While, Eigenfaces correctly identified 23 out of a total of 25 images, with only two image error, for a 92% accuracy and Fisherfaces mechanism identified 21 out of the overall 25 images with a error of four images and an accuracy of 84%. Fisherfaces mechanism accuracy is noticeably worse than the preceding two. The total number of images in datasets 2 and 3 has now been enhanced to 60 and 150, respectively. The accuracy of all three mechanisms considerably improves, with LBPH exhibiting the lowest amount of error (one image out of 150 total images) and an accuracy of 99.33%. Fisherfaces improved the most overall from datasets 1 and 2, obtaining a 96% and six image errors out of a total of 150 images. The Eigenfaces technique saw a slight error modification as compared to earlier datasets, obtaining an accuracy of 98.66%. All three models accuracy increases as the dataset increases but LBPH provides more accuracy on the same datatsets.

Eigenfaces and Fisherfaces mechanism has a few major disadvantages; the accuracy of Eigenface is affected by different lighting conditions. This may create a problem in real life as finding the best lighting conditions always is not possible. While the LBPH algorithm also recognizes faces from a slightly side perspective that is highly helpful for visually impaired people who are completely blind and have trouble while the the Eigenfaces algorithm demands that the face be precisely looked at the camera. Due to their vision impairment, our user may or may not be facing the camera directly, hence this is a crucial consideration. As a result, we choose Linear Binary Pattern Histograms over Eigenfaces. Dataset training is required to train the model with the user's data, as depicted in Fig. 3. In the face recognition model, the user's data is fed by training the model with as many more data as possible. The face and name of the user is detected in Fig. 4, it detects faces with eyes closed as well that will be of great help for few of the visually impaired. In the recent times there has been reservations concerning face recognition bias, this problem can be solved by feeding more and more data which help us reduce the bias.

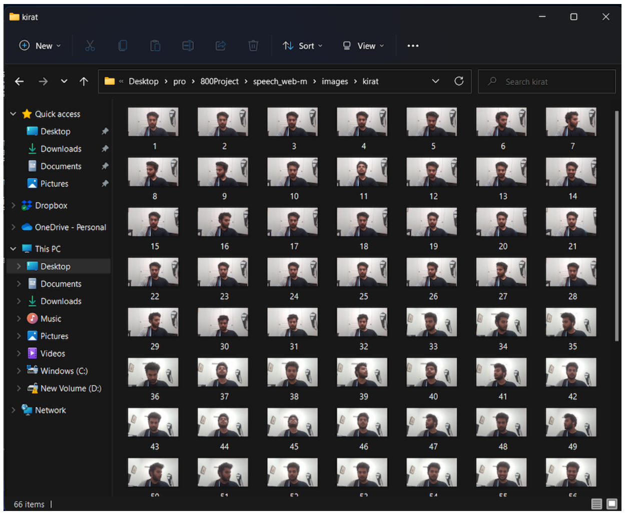


Fig. 3. Dataset preparation.

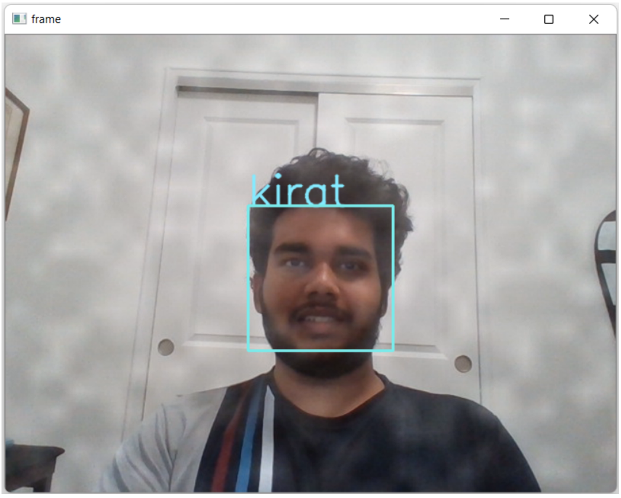


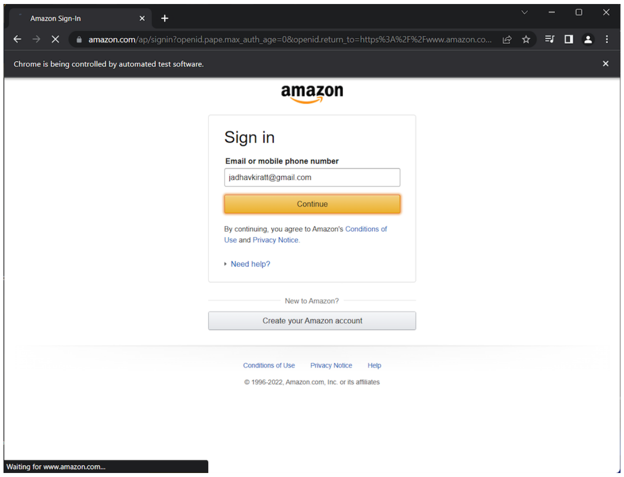
Fig. 4. Face recognition implementation.

D. Browser Automation

Automating the browser is one of the important aspect in this project. We use Google Chrome as the base browser. Automation is helpful in making the user's voice to function. There are many web driver tools, which helps in scrapping the data from the e-commerce website. The industry standard tool for test automation is Selenium Web Drivers. There were other alternatives framework like Scrappy, for web scrapping but here as we deal with automating the entire web browser, we use selenium web drivers. If we were just dealing with scrapping html files, Scrappy framework would be a great tool as well.

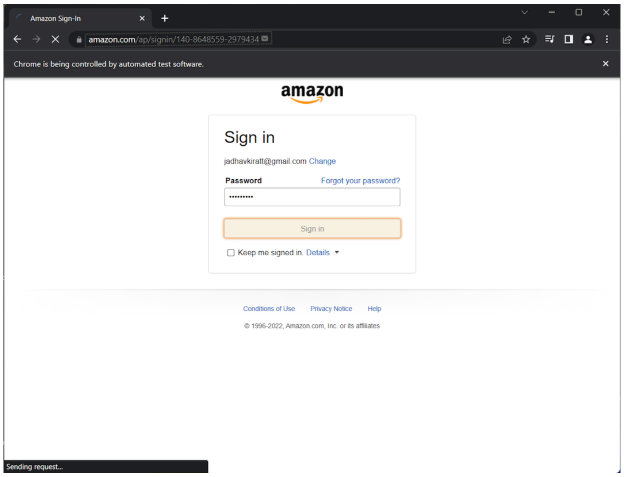
Selenium is a web based application used for writing test scripts in various languages. We the inspect element feature in chrome to take the exact element ID 'ap_email' of the textbox, button, etc. After the face recognition the part, the web driver directs the to the e-commerce website, in this case 'Amazon.com'.

As the email ID and password is saved previously in the code corresponding to the username. After face recognition, the browser is directed to the login page and using web scrapping the ID of the text box is taken and the email is inserted using the send_keys function and then with the help of .click() function the button is clicked. The same is depicted in Figs. 5 and 6.


Amazon Sign-In
amazon.com/ap/signin?openid.pape.max_auth_age=0&openid.return_to=https%3A%2F%2Fwww.amazon.co...
Chrome is being controlled by automated test software.
amazon
Sign in
Email or mobile phone number
Continue
By continuing, you agree to Amazon's Conditions of Use and Privacy Notice.
Need help?
New to Amazon?
Create your Amazon account
Conditions of Use
Privacy Notice
Help
© 1996-2022, Amazon.com, Inc. or its affiliates
Waiting for www.amazon.com...

jadhavkiratt@gmail.com

Fig. 5. Login email.


Amazon Sign-In
amazon.com/ap/signin/140-8648559-2979434
Chrome is being controlled by automated test software.
amazon
Sign in
Change
Password
Forgot your password?
Sign in
Keep me signed in. Details
Conditions of Use
Privacy Notice
Help
© 1996-2022, Amazon.com, Inc. or its affiliates
Sending request...

jadhavkiratt@gmail.com

Password: •••••••••

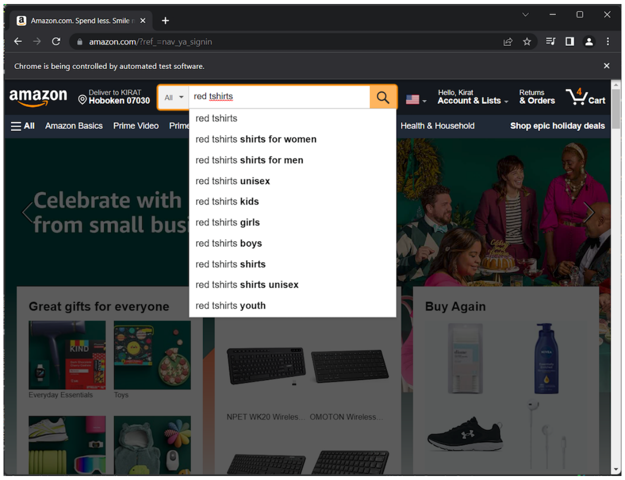
☐ Keep me signed in.

Fig. 6. Login password.

Later, through gTTS the code communicates with the user as in what product does he want to search. For example if he says Red Tshirts, the speech will be recognised and converted into text and added in the search box using selenium driver using element ID ”twotabsearchtextbox” and searched refer Fig. 7.

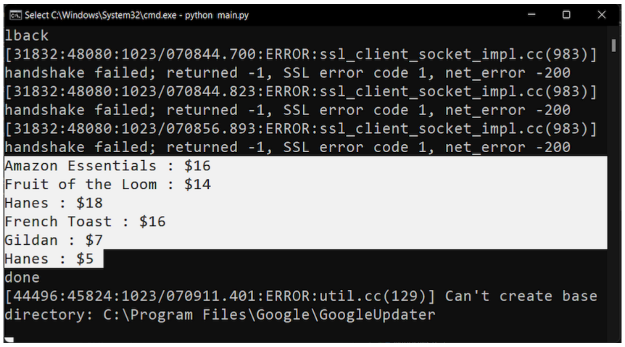
Next, once the search is complete and specified product is displayed. The software will run and filter the top 5 product based on the rating, here we used the element ID ’p_72/1318477031.’ The filtered product list is later scrapped and saved as highlighted in Fig. 8.

Later the the system will spell out the selections to the user through gTTS and once the user selects one product, the product will added to cart and later for checkout.



red tshirts

Fig. 7. Sample product search.



```
lback
[31832:48080:1023/070844.700:ERROR:ssl_client_socket_impl.cc(983)]
handshake failed; returned -1, SSL error code 1, net_error -200
[31832:48080:1023/070844.823:ERROR:ssl_client_socket_impl.cc(983)]
handshake failed; returned -1, SSL error code 1, net_error -200
[31832:48080:1023/070856.893:ERROR:ssl_client_socket_impl.cc(983)]
handshake failed; returned -1, SSL error code 1, net_error -200
Amazon Essentials : $16
Fruit of the Loom : $14
Hanes : $18
French Toast : $16
Gildan : $7
Hanes : $5
done
[44496:45824:1023/070911.401:ERROR:util.cc(129)] Can't create base
directory: C:\Program Files\Google\GoogleUpdater
```

Fig. 8. Filtered product.

IV. Conclusion

In this paper, we looked at the best approach for making online e-commerce more accessible to people with visual impairments. We examined various text-to-speech libraries to identify the best library for this case considering specific condition for visually impaired, gTTS and SpeechReconition libraries were selected for a smoother communication between the system and the consumer. These libraries were selected using the word error rates formula (lower the WER, the better) and formulated 13.2% and 16.5% word error rates that fall in the good scale. To provide a flawless experience for the visually challenged user, we studied and implemented different facial recognition mechanisms LBPH, Eigenfaces, and Fisherfaces. We executed this comparison using three different sizes of datasets, i.e., 25, 60, and 150 images. LBPH with Haar filter was chosen because it had the highest accuracy (99.33%) and it also allowed the customer to recognize their face from slightly side angles as well. While, the Eigenfaces and Fisherfaces too gave a good accuracy of 98.66% and

96% but its incapability of recognizing the video feed with a side angle was a demerit here as the consumer here is visually impaired, so the consumer directly looking into the camera is not guaranteed. For automating the browser, we choose Selenium web drivers for making the further process handsfree. There is a scope for future work like making the whole system portable and increasing the test data size.

Project GitHub link: https://tinyurl.com/43cf2m65

REFERENCES

[1] Ackland P, Resnikoff S, Bourne R. World blindness and visual impairment: despite many successes, the problem is growing. Community Eye Health. 2017;30(100):71-73. PMID: 29483748; PMCID: PMC5820628.

[2] J. Howse, ”Training detectors and recognizers in Python and OpenCV,” 2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2014, pp. 1-2, doi: 10.1109/ISMAR.2014.6948516.

[3] D. Petrović and I. Stanišević, ”Web scrapping and storing data in a database, a case study of the used cars market,” 2017 25th Telecommunication Forum (TELFOR), 2017, pp. 1-4, doi: 10.1109/TELFOR.2017.8249451.

[4] P. Viola and M. Jones, ”Rapid object detection using a boosted cascade of simple features,” Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[5] Ali, A. A., El-Hafeez, T. A., Mohany, Y. K. (2019). An Accurate System for Face Detection and Recognition. Journal of Advances in Mathematics and Computer Science, 33(3), 1-19. https://doi.org/10.9734/jamcs/2019/v33i330178.

[6] P. Ramya, V. Sindhura, and P. V. Sagar, ”Testing using selenium web driver,” 2017 Second International Conference on Electrical, Computer and Communication Technologies (ICECCT), 2017, pp. 1-7, doi: 10.1109/ICECCT.2017.8117878.

[7] K. Lee, C. Lee, S. -A. Kim, and Y. -H. Kim, ”Fast object detection based on color histograms and local binary patterns,” TENCON 2012 IEEE Region 10 Conference, 2012, pp. 1-4, doi: 10.1109/TENCON.2012.6412323.

[8] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, ”Eigenfaces vs. Fisherfaces: recognition using class specific linear projection,” in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711-720, July 1997, doi: 10.1109/34.598228.

[9] G. G. Dordinejad, and H. Çevikalp, ”Face Frontalization for Image Set Based Face Recognition,” 2022 30th Signal Processing and Communications Applications Conference (SIU), 2022, pp. 1-4, doi: 10.1109/SIU55565.2022.9864911.

[10] A. U. Naik, and N. Guinde, ”LBPH Algorithm for Frontal and Side Profile Face Recognition on GPU,” 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020, pp. 776-779, doi: 10.1109/ICSSIT48917.2020.9214228.

[11] F. A. Nazira, M. I. Uddin, M. H. Raju, S. Hossain, M. N. Rahman, and M. F. Mridha, ”Face Recognition Based Driver Detection System,” 2021 International Conference on Data Analytics for Business and Industry (ICDABI), 2021, pp. 572-577, doi: 10.1109/ICDABI53623.2021.9655889.

[12] K. -H. Chan, and C. -M. Chao, ”DriverID: Driver Identity System Based on Voiceprint and Acoustic Sensing,” 2022 IEEE International Conference on Consumer Electronics - Taiwan, 2022, pp. 45-46, doi: 10.1109/ICCE-Taiwan55306.2022.9869000.

[13] F. Ming, Z. Zhou, and Z. Li, ”The design and implement of the cross-platform mobile automated testing framework,” 2016 5th International Conference on Computer Science and Network Technology (ICCSNT), 2016, pp. 182-185, doi: 10.1109/ICCSNT.2016.8070144.