

**COMSATS University**

**Islamabad Sahiwal, Pakistan**

**<** **Audio-visual speech recognition >**

***By***

**Kainat Iqbal CIIT/FA18-BCS-021/SWL**

**Ahmad Raza CIIT/FA18-BCS-005/SWL**

**Usama CIIT/FA18-BCS-048/SWL**

***Supervisor***

**Dr. Azhar**

***Bachelor of Science in Computer Science (2018-2022)***

**The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others**

**COMSATS University**

**Islamabad, Sahiwal Pakistan**

**<Audio-visual speech recognition>**

**A project presented to**

**COMSATS University Islamabad, Sahiwal Campus**

**In partial fulfillment**

**Of the requirement for the degree of**

***Bachelor of Science in Computer Science (2018-2022)***

**By**

**Kainat Iqbal** **CIIT/FA18-BCS-021/SWL**

**Ahmad Raza** **CIIT/FA18-BCS-005/SWL**

**Usama** **CIIT/FA18-BCS-048/SWL**

**DECLARATION**

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely based on our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

**Kainat Iqbal Ahmad Raza Usama**

**---------------------- ---------------------- ---------------------**

**CERTIFICATE OF APPROVAL**

It is to certify that the final year project of BS (CS) “Audio-Visual Speech Recognition” was developed by **Ahmad Raza (CIIT/FA18-BCS-005/SWL), Kainat Iqbal (CIIT/FA18-BCS-021/SWL)** and **Usama (CIIT/SP18-BCS-048/SWL)** under the supervision of **“Dr Azhar”** that in their opinion; it is fully adequate, in scope and quality for the degree of Bachelor of Science in Computer Sciences.

---------------------------------------

**Dr. Azhar**

Supervisor

Associate Professor

Department of Computer Science

COMSATS University Islamabad, Sahiwal Campus

----------------------------------------

**External Examiner**

Lecturer

Department of Computer Science

COMSATS University Islamabad, Sahiwal Campus

---------------------------------------

**Head of Department**

(Department of Computer Science)

**Acknowledgement**

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor **Dr. Azhar** Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are deeply indebted to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

**Kainat Iqbal Ahmad Raza Usama**

**---------------------------- ---------------------------- ---------------------------**

**Executive Summary**

Deep lip reading is a new concept in the field of artificial intelligence. Among the most exciting aspects of this research is that it combines two of the most innovative branches of the domain, namely, computer vision and natural language processing. In addition, as with all deep learning technologies, it is plagued by a black-box problem. Consequently, understanding the inner workings of the networks is extremely difficult, especially concerning lip lip-reading; the network architectures are often even more complicated and combine several different technologies. The project aims to perform a sensitivity analysis of lip reading with Lip Net a (deep neural network for visual speech recognition) architecture to understand the architecture and improve the understanding of deep lip-reading processes

In our daily life we deliver thoughts and thinking either by writing or speaking. In this fast-paces life where people are connected through internet as compared to communication through face to face. So, it is difficult for the people to communicate in noisy environment like in factory areas and those areas where internet connectivity is very week during the video conferencing call. So, we have decided to make a system which help people regarding these issues so they can communicate smoothly.

The important issue is that a person who had lost his voice accidently or those people who are unable to hear. It is difficult for those people to communicate and for normal peoples to understand what they are saying. so, we have decided to provide a platform for those people to make their communication smooth so they can send and receive messages.

|  |  |
| --- | --- |
| **SRS** | Software Requirement System |
| **SDD** | Software Design Description |
| **DAPP** | Decentralized Application |
| **IPFS** | Interplanetary File System |
| **JSX** | JavaScript Extensible Markup Language |

**Abbreviations**

Table of Contents

[**1 Introduction 1**](#_Toc89972460)

[1.1 Brief 1](#_Toc89972461)

[1.2 Relevance to Course Modules 1](#_Toc89972462)

[1.3 Project Background 2](#_Toc89972463)

[1.4 Literature Review 2](#_Toc89972464)

[1.5 Analysis from Literature Review 2](#_Toc89972465)

[1.6 Methodology and Software Lifecycle for This Project 2](#_Toc89972466)

[1.7 Rationale behind Selected Methodology 3](#_Toc89972467)

[**2** **Problem Definition** 5](#_Toc89972468)

[2.1 Problem Statement 5](#_Toc89972469)

[2.2 Deliverables and Development Requirements 5](#_Toc89972470)

[2.3 Current System 6](#_Toc89972471)

[**3** **Requirement Analysis** 8](#_Toc89972472)

[3.1 Use Cases Diagram 8](#_Toc89972473)

[3.2 Functional Requirements 13](#_Toc89972475)

[3.3 Non-Functional Requirements 14](#_Toc89972476)

[3.3.1 Usability 14](#_Toc89972477)

[3.3.2 Performance 14](#_Toc89972478)

[3.3.3 Security 15](#_Toc89972479)

[**4** **Design and Architecture** 17](#_Toc89972480)

[4.1 System Architecture 17](#_Toc89972481)

[4.2 Process Flow/Representation 18](#_Toc89972482)

[4.3 Class Diagram 19](#_Toc89972483)

[4.4 Sequence Diagram 19](#_Toc89972484)

[4.5 Design Models 24](#_Toc89972485)

[4.5.1 Data Flow Diagram 24](#_Toc89972486)

[**5** **Implementation** 28](#_Toc89972487)

[5.1 Algorithm 28](#_Toc89972488)

[5.2 User Interface 28](#_Toc89972489)

[**6** **Testing and Evaluation** 32](#_Toc89972490)

[6.1 Manual Testing 32](#_Toc89972491)

[6.2 System testing 32](#_Toc89972492)

[6.3 Unit Testing 32](#_Toc89972493)

[6.4 Integration Testing 33](#_Toc89972494)

[**7** **Conclusion and Future Work** 36](#_Toc89972495)

[7.1 Conclusion 36](#_Toc89972496)

[7.2 Future Work 36](#_Toc89972497)

[7.3 References 37](#_Toc89972498)

List of Figures

[Figure 1: Use Case Systems 8](#_Toc89973946)

[Figure 2: Use Case Login/Sign up 9](#_Toc89973947)

[Figure 3: Use Case instructor Module 10](#_Toc89973948)

[Figure 4: Use Case Enroll Courses 11](#_Toc89973949)

[Figure 5: Use Case Chatbot 12](#_Toc89973950)

[Figure 6: Architecture 17](#_Toc89973951)

[Figure 7: Class Diagram 19](#_Toc89973952)

[Figure 8: Sequence Diagram (Login) 20](#_Toc89973953)

[Figure 9:Sequence Diagram (Request Approval) 21](#_Toc89973954)

[Figure 10: Sequence Diagram ( Create Course) 22](#_Toc89973955)

[Figure 11: sequence diagram (Courses Module) 23](#_Toc89973956)

[Figure 12: System overview 24](file:///E:\FYP\Final%20Report(FYP).docx#_Toc89973957)

[Figure 13: User Interface( Login) 28](#_Toc89973958)

[Figure 14:User Interface (Register ) 28](#_Toc89973959)

List of Table

[Table 1: Deliverables and Development 5](#_Toc89973976)

[Table 2:Detailed Use Case(System) 9](#_Toc89973977)

[Table 3:Detailed Use Case(Login/Signup) 10](#_Toc89973978)

[Table 4:Detailed Use Case(Create Courses) 11](#_Toc89973979)

[Table 5:Detailed Use Case(Courses Enrollment) 12](#_Toc89973980)

[Table 6:Detailed Use Case(Chatbot) 13](#_Toc89973981)

[Table 7:Verification of Instructor 13](#_Toc89973982)

[Table 8:Uploading Courses 13](#_Toc89973983)

[Table 9:Courses Enrollment 14](#_Toc89973984)

[Table 10:Chatbot 14](#_Toc89973985)

[Table 11:Unit Testing 1 32](#_Toc89973986)

[Table 12: Unit Testing 2 32](#_Toc89973987)

[Table 13: Integration Testing 1 33](#_Toc89973988)

[Table 14: Integration Testing 2 33](#_Toc89973989)

[Table 15: Integration Testing 3 34](#_Toc89973990)

[Table 16: Integration Testing 4 34](#_Toc89973991)

[Table 17: Integration Testing 5 34](#_Toc89973992)

# **Introduction**

## **1.1. Project Brief**

Communication is essential for human life and is a fundamental part of human contact and interaction with other humans. We have many communication methods that can be generalized into verbal, non-verbal and visual communication. Lipreading can combine both verbal and non-verbal actions of communication. Lipreading, also known as visual language recognition, is a way of visually interpreting the lip movement of people by looking at their faces to recognize what is being said. It is a way to visually “listen” to people, therefore often referred to as “a third ear”. Humans use it naturally to improve speech understanding though it is not yet clear how our brains manage to perform those actions

Lip reading is a technique which is used to understand words or speech by visual interpretation of face, mouth, and lip movement without the involvement of audio. It is helpful for the people who are unable to hear we have many problems in sending and receiving messages. Due to lack of network problem and ping some of the signals are drop and the required message is disturbed. Our system will provide help in communication by converting the movement of lips into text through scanning the lips.

lip-reading is considered as the only viable solution to the problem of delivering cleared massage from one person to another.

### Relevance to Course Modules

We will use feature extraction in our project which is normally refers to the process of extracting features (informative characteristics) from a frame in a video, independently of past or future frames. I would say that it is very similar to extracting features in a static image. Maybe there exist methods that use several frames to extract more stable features or something like that. This process, for example, is usually (not exclusively) done to initialize an object tracking method to recognize the object.

There will be many components that will need to be reused in different places, so we will try to write code that will be reusable in most cases. There are two parts to this project e.g., front-end and back-end. We will try to apply this methodology in both places.

With the rise in computer power and a vast amount of benchmark data availability, we can train the computer to recognize the lip movements and predict what has been spoken by applying different machine learning and deep learning algorithms. We can use robust deep neural networks like CNNs and RNNs with many benchmark architectures like Alex Net and Res Net, which helps us get better performances by reusing them. Deep Learning, a subcategory of machine learning, can mimic the function of the human brain, also known as artificial neural networks (ANN)

In the end we will use classification to predict the actual spoken words, classification refers to a predictive modeling problem where a class label is predicted for a given example of input data. Examples of classification problems include. Given an example, classify if it is spam or not.

## **Project Background**

# This project focus on developing an artificial intelligence module which provide the facility to recognize movement of lips and converted it into the text without the involvement of audio. As people use different dictions and various ways to articulate a speech so it is very difficult to devise an automated lip-reading system. Our purpose is to make effective communication in noisy environment, excellent video call conferencing and online meeting and easy to communicate with faulty people. There are some platforms but not enough qualified and reliable to be use. So, considering all above scenarios in our mind we have design a system that is qualified for all above statements.

## **Literature Review**

Although lip reading has a long history of its own and getting used from 1500 A.D. to the present, however, Deep lip reading is a new technique in the field of deep learning. DLR is a subset of Deep learning that attempts to simulate the human brain's behavior by learning from substantial amounts of data using the NN. Deep knowledge has provided a driving force to many services and products we use daily, such as credit card fraud detection, digital assistants, face recognition, sentiment analysis, and emerging technologies like self-driving cars, et cetera. DLR itself uses Deep learning technology and provides and can provide a wide range of application values in public safety, visual synthesis, and particularly grasping human speech for the listeners with hearing impairment. DLR can serve deaf and hard of hearing people who interact with others and do not know sign language (Zhang et al., 2021). Moreover, it can help interact in a noisy environment and assist patients with throat injuries or vocal cord troubles. Furthermore, it can help analyses CCTV video footage and provide insights to the police or similar scenarios to find out what someone is saying and what is happening. Deep Lip reading

can consist of audio and visual components or only visual components without audio. Therefore, it provides the scope to research natural language processing (NLP) and computer vision. The latter has seen significant improvement, especially with the adaption of Convolutional Neural Network (CNN) architectures such as Alex Net, VGG Net, GoogLe Net, Res Net, Dense Net because of its visible nature.

Different face tracking libraries are available, such as dlib (), face alignment etc., to locate the lip region, i.e., the region of interest. Autoencoders and especially Convolutional Neural Networks (CNNs) are used in frontends. In recent times CNNs are the most used frontend networks due to their learning supremacy on spatial and temporal features and extracting more relevant features. Lip-reading predicts speech sequential in nature for the 7 backend, like words or sentences. Researchers tend to use Recurrent Neural Networks (RNNs). However, in recent years, TCNs and attention-based Transformers have been used instead of Long-Short Term Memory networks (LSTMs) or Gated Recurrent Units (GRUs). Architectures using Attention-Transformers and TCNs seem to have advantages over RNNs when used as backend.

## **Analysis from Literature Review**

In Pakistan there is not a proper lipreading application especially for deaf and dumb for students of matric and intermediate level students there is a need of such platform where students can learn from the instructors of their own choice.

After we tried and tested some of the system, we came to know that there is no system available that provides the facility both for students and instructors to learn under one platform with use of technology. This made us clear that there is no app in the market that performs these functions, or if there is then it is not visible enough to get the attention.

## **Methodology and Software Lifecycle for This Project**

The design methodology that we will use to develop this software will be Object-Oriented Programming (OOP). We are choosing this because we can the many benefits of OOP in developing our system. The most important one is Reusability. We have different modules in our application, and we don’t want to build everything from scratch for a new module.

For this project we adopted “Incremental software process model.” As the incremental build model is a method of software development where the product is designed, implemented and tested incrementally until the product is finished. It involves both development and maintenance. We will deploy project time by time to check bugs to fix it and improve performance and make more usable to increase usability.

## **1.7** **Rationale behind Selected Methodology**

We will use **“Incremental Developmental approach”.** This is because we have classified our project requirements into stand-alone four modules. In this Software Development Life Cycle (SDLC), there is four parts Requirement analysis, designing, coding, and Testing. This cycle requires good design planning but throughout of the development system changes can be done on each stage. This model is less costly as compares to other. In this system complexity generated when you rectify the problem in one stage it requires correction in all stages. It consumes a lot of time. In incremental model, we can reduce the complexity of the system by working throughout on the requirement of the client and prioritized its requirement. Thus, Incremental model is best approach to handle the system.

**Chapter 2**

**Problem Statement**

# **Problem Definition**

## **Problem Statement**

Our system can serve deaf and hard of hearing people who interact with others and do not know sign language. Moreover, it can help interact in a noisy environment and assist patients with throat injuries or vocal cord troubles. Furthermore, it can help analyze CCTV video footage and provide insights to the police or similar scenarios to find out what someone is saying and what is happening. Deep Lip reading can consist of audio and visual components or only visual components without audio. Therefore, it provides the scope to research natural language processing (NLP) and computer vision

The main problem is to communicate in noisy areas. Like in factory areas the workers cannot convey their visual messages The important issue is that a person lost voice accidently. During the video conferencing call, or any online meeting due to weak internet signals voice often drops. The result is judged based on accuracy. In this system we try to improve the accuracy of the lip movement of the previous systems by using AI modules. We will get knowledge about the algorithms of AI and about the python libraries.

## **Deliverables and Development Requirements**

Table 2.1: Deliverables and Development

|  |  |  |  |
| --- | --- | --- | --- |
| **Tools** | **Tools** | **Versions** | **Rationale** |
| **&** | MS Studio | 2015 | IDE |
| **Technologies** | PyCharm | 2016 | Development |
|  | MS Word | 2013 | Documentation |
|  | MS PowerPoint | 2013 | Presentation |

## **Current System**

Following applications that are currently working in Pakistan:

* Lip Reading in the Wild
* Lipreading using Temporal Convolutional Networks

Table 2.2: Current System

|  |  |  |
| --- | --- | --- |
| **Application Name** | **Weakness** | **Proposed Project Solution** |
| Lip Reading in the Wild | Shortage of suitable datasets. | Automated large-scale data collection from TV broadcasts |
| Lipreading using Temporal Convolutional Networks | the current state-of-the-art methodology produces models that do not generalize well to variations on the sequence length | We address this issue by proposing a variable-length augmentation |

**Chapter 3**

**Requirement Analysis**

# **Requirement Analysis**

A use case diagram is a way of summarizing details of a system and the users within that system. How users will perform tasks on our application. It outlines, from a user’s point of view, a system’s behavior as it responds to a request. Each use case is represented as a sequence of simple steps, beginning with a user's goal, and ending when that goal is fulfilled.

A use case diagram is a graphic depiction of the interactions among the elements of a system. Use case is a methodology used in system analysis to identify, clarify, and organize system  
requirements. In this context, the term "system" refers to something being developed or operated.

## **Use Cases Diagram**

## **Use case 01(Audio-visual speech recognition)**

Diagram

Description automatically generated

**Figure 3.1:** *Use Case (*Audio-visual speech recognition)

**Detailed Use Case**

Table3.1 Detailed Use Case (Audio-visual speech recognition)

|  |  |
| --- | --- |
| **Use Case ID:** | UC1 |
| **Use Case Name:** | Audio-visual speech recognition |
| **Actors:** | Primary Actor: User |
| **Description:** | People who are deaf and dumb access the app and get their work done easily. It will provide a quick way for them to get their message delivered quickly and efficiently. |
| **Trigger:** | A User indicates that they want to deliver their message by some mean of communication, through our app. |
| **Preconditions:** | PRE-1. User must log into the app before using it  . |
| **Post conditions:** | POST-1. User will get his video into text.  POST-2. He can also save it. |
| **Normal Flow:** | User has to login for profile by their username and Password.  Then upload the video and wait for processing.  Our app for now only provided the option of video to text. |
| **Alternative Flows:** | User can convert his video into text.  After processing and getting the result, the user can logout. |
| **Exceptions:** | Camera not working |
| **Business Rules:** | BR-1 The video must be completed in two to three minutes |

**Use Case 02: Video to text**

Diagram

Description automatically generated

**Figure 3.2:** Use Case Video to Text

**Detailed Use Case**

Table 3.2 Detailed Use Case (Video to text)

|  |  |
| --- | --- |
| **Use Case ID:** | UC2 |
| **Use Case Name:** | Video to text |
| **Actors:** | Primary Actor: User |
| **Description:** | The video will be sent for the processing in the back end.  After processing it will show in text format.  After that the user can also go for the new video  User can also convert recorded video by uploading it on to the processor and then it will convert into text  User can also save converted video |
| **Trigger:** | A User want to convey his message, through our app. |
| **Preconditions:** | PRE-1. System should be trained on regarding dataset.  . |
| **Post conditions:** | POST-1. He also can share it.  POST-2. User will get his video into text.  POST-3. He can also save it. |
| **Normal Flow:** | Extract, find out, match the feature or object in one frame of the video.  Class label is predicted for a given example of input data. |
| **Alternative Flows:** | User can convert his video into text.  After processing and getting the result, the user can logout. |
| **Exceptions:** | Camera not working |
| **Business Rules:** | BR-1 Processing should be completed in two to three minutes |

## **Functional Requirements**

This section describes the functional requirements of the system expressed in natural language style. This section is typically organized by feature as system feature name and specific functional requirements associated with this feature

**Video inserting**

Table 3.3 Video inserting

|  |  |
| --- | --- |
| **Identifier:** | FR-2 |
| **Title:** | Video inserting |
| **Requirement:** | User shall be able to upload the video from which he wants to read lip movements |
| **Source:** | FR-1 will originate it. |
| **Rational:** | The user can input the video from gallery which he already recorded. It will also provide him freedom to import from phone storage as well |
| **Dependencies:** | FR-1 |
| **Priority:** | High |

**Video to text**

Table 3.4 Video to text

|  |  |
| --- | --- |
| **Identifier:** | FR-3 |
| **Title:** | Video to text |
| **Requirement:** | User shall be able to covert any video either recorded or saved video saved in device |
| **Source:** | FR-2 will originate it. |
| **Rational:** | the video will be sent for the processing in the back end. After processing it will show in text format. After that the user can also go for the new video |
| **Dependencies:** | FR-2 |
| **Priority:** | High |

## **Non-Functional Requirements**

Non-Functional Requirements are the requirements that specify criteria that can be used to judge the operation of system. Those constraints under which system will be operated is called non-functional requirements. For example, language run time environment, operating environment performance requirements, usability requirements etc. These are all those requirements whose are not belonged to functional requirements, but they affect overall on the system. We can say some extra conditions requirements that are not included in the use cases. These are usually called non-functional requirements; some of these are given below.

### **Usability**

The system will help in saving time and improving efficiency. This system will give the facility to help both defected and normal users. Clear and high-quality function will be carried out by the system.it maximize the utility of information encoded in different facial regions.

### **Performance**

Performance of the system will be excellent. It will give every conversion very correctly and provide all the results with efficiency and accuracy. It will not slow-down the system even at peak hours without affecting the quality of service of the system even with poor internet connection.

### **Ease of use**

The working of the system will be very easy. The user just opened the system and sent the input either by recording the live video or insert from the system. After the insertion of the video the system process on that input and gives the output.

**Chapter 4**

**Design and Architecture**

# **Design and Architecture**

## **System Architecture**

The image below describes the system architecture. It mainly has two components, which are backend system and frontend interface. The backend processes the input video and performing required operations. The structure of the system explains its core components, their relationships, and how they deal with each other. In Design, functional requirements are accomplished. Anyone can visit, anyone can visit the main page related to system and can get information and know about our system that how to use to system and how anyone can give their specific video. A combination of the modules makes up the system.

Diagram

Description automatically generated

**Figure 4.1 :** Architecture

## **Process Flow/Representation**

In flow chart representation we gave a detail view of functionalities of our complete system, user request and the response of the system to that request. It shows what functionalities can perform in our system and how our system will answer queries of a user.

Diagram

Description automatically generated

**Figure 4.2:**  Process Flow Diagram

## **Sequence Diagram**

Sequence Diagrams are interaction diagrams that detail how operations are carried out. This shows that how system will work, tasks after task with sequence wise. It gives the complete description the working of our system.

* **Diagram

  Description automatically generatedSequence Diagram 01: Insert video**

**Figure 4.3:** Sequence Diagram (Insert video)

In figure 8 the flow of video inserting function for registered users represents. The user can insert the video from which he wants to read lips movement

**Sequence Diagram 02: Video into text**

**Diagram

Description automatically generated with low confidence**

**Figure 4.5**: Sequence Diagram (Video into text)

In Figure 10 the flow of video into text function for registered users the video will be

sent for the processing in the back end. After processing it will show in text format. After that the user can also go for the new video

## **Design Models**

A design model is an object-based picture or pictures that represent the use cases for a system. Or to put it another way, it's the means to describe a system's implementation and source code in a diagrammatic fashion.

### **Data Flow Diagram**

**LEVEL 0:**

Diagram

Description automatically generated

**Figure 4.6:** : System overview

**Chapter # 5**

**Implementation**

# **Implementation**

## **Algorithm**

Uses two algorithms to match intents: CNN and LSTM. Our first model ran every image of our sequenced input through a Convolutional Neural Network and then fed the flattened outputs as a sequence into a Long Short-Term Memory Recurrent Neural Network, which produced a single output, making it a many-to-one RNN. The LSTM was added to package the entire sequence of CNN outputs into a single layer without losing the temporal understanding of the video frames. An LSTM fixes the vanishing gradient problem present in vanilla RNNs, which inhibits the backpropagation of gradients to occur. It does so by adding 4 gates (input (i), forget (f), output (o), new memory (c)) whose activations can be learned, to control whether to hold on to information: Given that we use SoftMax as our last activation, our loss function is cross entropy loss: Finally, we used the Adam Optimizer to better navigate through the loss function.

## **User Interface**

A member can get access directly by running the system into their personal computer. Then he/she can use this service for their desired purpose. A member upload video from the device or can this use service on live video

**Chapter6**

**Testing and Evaluation**

# **Testing and Evaluation**

**6.1 Manual Testing**

Manual testing is the process of finding out defects, errors and bugs in our software application and this testing verifies that all the features are properly working or not. Tester manually executes the test cases. Manual testing is the process of using the features of an application as an end-user. Considering the scope of the project and the time limitations, we have performed following tests.

## **System testing**

Once the system has been successfully developed, testing must be performed to ensure that the system working as intended. This is also to check that the system meets the requirements stated earlier. Besides that, system testing will help in finding the errors that may be hidden from the user. There are few types of testing which includes the unit testing, functional testing, and integration testing. The testing must be completed before it is being deploy for user to use.

## **Unit Testing**

**Unit Testing 1:** Recorded the Video.

**Objective:** To ensure that the video can be recorded or not

Table 6.1: Unit Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No**. | **Test case/Test Script** | **Attribute and**  **Value** | **Expected Result** | **Result** |
| 1. | Video recorded | User image or user video | Video recorded successfully | Recorded then saved |

## **6.4 Functional Testing**

After successful unit testing, we are here to check functional testing. Functional testing is the stage in which we test all the modules of the system.

**Functional Testing 1:**

**Table 6.2: Unit Testing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test case/Test script** | **Attribute and value** | **Expected result** | **Result** |
| 2. | Recording Video From mobile camera and receive sentence/word from Server | Camera | Camera functions correctly | Pass |

**Functional Testing 2:**

**Table 6.3:** Unit Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test case/Test script** | **Attribute and value** | **Expected result** | **Result** |
| 3. | Receive video from front end and sending it to the image processing then receiving sentence or word from neural network and send it to the front end | Video and converted text | Video is converted into correct text | Pass |

**Functional Testing 3:**

**Table 6.4:** Unit Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test case/Test script** | **Attribute and value** | **Expected result** | **Result** |
| 4. | Video saved into inference history | Inference history | saved | Pass |

## 

## **6.5 Integration Testing:**

**Table 6.5:** Integration Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test case/Test script** | **Attribute and value** | **Expected result** | **Result** |
| 1. | Video of speaker move to the image processing unit for extraction of lips movement | Video from user | Words of each speaker | Pass |
| 2. | Inset video of the user from their device | Video | Video is inserted successfully | Pass |
| 3. | Save converted video | History | Successfully saved | Pass |

## **Automated Testing:**

A lot of professional experience is required for automatic testing and the tools are also not available for free. To check if the system is working correctly or not. Hence, we used manual testing.

**Chapter 7**

**Conclusion and Future Work**

# **Conclusion and Future Work**

## **Conclusion**

A project review reveals that a relatively more significant portion of time has been spent exploring and running the lip-reading TCN architecture rather than model implementation to visualize the architecture for understanding lip-reading neural network interpretation. We wanted to work on an end-to-end lip-reading in the initial project plan. However, considering the lack of background knowledge of the author in the respective field, time constraints and the complexity of the work pushed the author to keep that task as a future work for further research. If any additional work is to be conducted in this area or other projects of similar nature are to be undertaken, this would be a critical factor to consider. To enhance the interpretability of the system and provide users with the ability to create and visualize network architectures for themselves, the author plans to continue working on the project. As mentioned in the previous section, several enhancements and enhancements are planned for future versions.

If we look in to last twenty years, almost every article focuses on illustrating the benefits of audio-visual speech recognition, but no serious attempt was made into creating this system expect Lip Net. We tried our best to provide a platform with accuracy to help peoples.

In conclusion, the problem facing people regarding hearing and speaking are numerous and with the current waves of public awareness, we utilize our engineering education in providing them a platform for reducing communication gaps. In beginning we start with 10 words and 100 sentences afterward we increase it and at the end there are almost 2 million words which user can utter by using our vocab and can get their results with higher accuracy.

## **Future Work**

Considering the project’s initial idea, the size was huge and complex for the author. There was a lack of background knowledge on the topic; therefore, there was a significant amount of time spent learning and researching artificial intelligence. The objective of performing sensitivity analysis is to complete and develop scripts to utilize the model to predict the user’s words or existing test video samples. There is plenty of tasks to perform in the future to extend this project and host remote user access to the demo system. The following improvements to the system are planned versions: -

* Now this system can only predict through lips movement but in future this will be able to predict through sign languages to compensate the confusion that occurs through lips movement alone. In future we will focus on adding more words in our vocabulary to help diversify speech.
* Now the greater limitation in our system is that it can only predict lip movement in only single face pose but in future through heavy deal of pre-processing prediction can be possible in every face side if they are obscured in any way.
* The applications are endless and are not just limited to the hearing. One of the possible uses of this app is that you may be able to look at your phone and mouth a command to your personal assistant app without speaking. Or you might point your phone’s camera at someone in a noisy room and have what they are saying dictated directly into your hearing aids via Bluetooth.
* Add more data for training the architecture to check the performance changes.
* Host the demo system so that users can remotely interact with the system.