***Table of Content:***

***1. Introduction……………………………………2***

***1.1 Purpose of Document…………………………2***

***1.2 Project Overview………………………………2***

***1.3 Scope……………………………………………3***

***2. Overall System Description……………………3***

***2.2 Operating Environment………………………3***

***2.2.1System constraints…………………………...4***

***2.2.2Software constraints………………………….4***

***2.2.3Hardware constraints………………………...4***

***2.2.4Cultural constraints……………………………4***

***2.2.5Legal constraints………………………………4***

***2.3 Assumptions and Dependencies ……………….5***

***3. Functional and Non-Functional requirements….5***

***4. External Interface Requirements…………………6***

***4.1 Hardware Interface………………………………6***

***4.2 Software Interface………………………………6***

***5. Use-Case………………………………………….6***

***6. References…………………………………………9***

***Introduction:***

***Purpose of Document:***

*The purpose of this document is to provide detail description of final year Project i.e., “Sign Language Recognition”. The main purpose would be to put up discussion between signers and non-signers. It describes various interfaces of the SLR, its operations, hardware and software requirements and its functionality. This document also contains functional and non-functional requirements of the project, constraints, assumptions and dependencies under which we will work. Functional requirements represent in details so we can easily understand the basic workflow and the activities that will help us in this document.*

***Project Overview:***

*This “Sign Language Recognition” focuses on complete explanation of the system. We also describe the specific requirements of the system. Our goal is to provide best system to communicate with deaf/dumb People by using sign language. To develop machine learning based on CNN that processes and classifies the sign language. Develop algorithm based on machine learning to detect and segment hand gesture in online. Design the structure of Sign language without use of any software hardware. Design algorithm of CNN based deep learning for detection, identification of sign language. Click image and perform transfer of gesture into sign language. At the end implantation and testing of our system.*

***Scope:***

*Sign Language Recognition project is planned to be used by speechless people in order to ease their life. This application work when the person stand in front of camera and his/her gesture are recognized by the camera. Camera specifies the special point on the body when user stand in certain position in order to track his/her body. After tracking body positions it provide output what it sees in front of its camera in the form of orientations of joints on the body.*

*And then match the output with pre-defined gesture and put output in the database. In this project it is aimed to recognize Pakistani Sign Language and translate it into text format. This program will work on PC environment.*

***Overall System Description:***

*This part gives information about product perspective, product functions and constraints, assumptions and dependencies respectively.*

***Operating Environment:***

*PSL will serve to speech-impaired people to learn sign language easily and not to have difficulty in communication by providing their gestures translating to the text for a person who does not know PSL.*

*The statement module is for translating gestures to the text. In terms of hardware, only PC and Camera is necessary. Gestures will be captured by Camera so no image processing tasks required. Our product aims to work in PC environment.*

*In terms of software, the constraints are:*

*• Operating System: Windows 7 or above*

*• Programming Language: Python and OpenCV*

***System Constraints:***

*Its main functionality is communication by using sign language and translates the text into English:*

*Its stages are:*

* *Catching the body movement*
* *Correlating gestures with the consistent words/phrases*

***Hardware Constraints:***

*Camera of**At least 5 MP. It is used to record the gesture of user.*

*RAM At least 4 GB or higher.*

*Processor of 2.6GHz or faster.*

***Software Constraints:***

*In our project all the coding is done in Python and OpenCV.*

***Cultural Constraints:***

*Only English is used for interaction with the user and display the output of gesture in English so it is easy to understand.*

***Legal Constraints:***

*There should be secure environment for the user and all the laws and regulation are followed and output is displayed correctly.*

***Assumptions and Dependencies:***

*In order to preserve the correctness, following assumptions are made:*

* *The user should stand between 1.8 to 2.4 meters away from the Camera.*
* *The Camera should be properly set up according to its guide.*
* *The body chasing phase should be properly done by the user.*
* *The program will work on PC environment mainly.*

***Functional Requirements:***

***Cross Platform support:***

*Software should run on as many platforms are available.*

***Authentic representation:***

*Software should give out correct meaning of gesture.*

***Gesture recognition:***

*Software should automatically recognize the gesture through video input.*

***Non-Functional Requirements:***

***Availability:***

*The software should be available all the time 24 hour a day.*

***Reliability:***

*The software should give correct meaning of gesture and easy to use.*

***Scalability:***

*The software should be able to handle all the gesture correctly without any disruption.*

***Maintainability:***

*The software should be coded in a way which is easily readable and maintainable.*

***External interface requirements:***

*This section is intended to specify any requirements that ensure that new system will connect properly to external components.*



***Hardware Interfaces:***

***Camera:*** *At least 5 MP. It is used to record the gesture of user.*

***RAM:*** *Al least 4 GB or higher.*

***App Server:*** *provide gesture recognition and server interaction.*

***Processor:*** *2.6GHz or faster.*

*HDD: 2GB available space.*

***Software Interfaces:***

*Operating system: Windows 7 or above*

*Programming Languages: Python and OpenCV used in our project for coding*

***Use-Case and description:***



*Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions that some systems (subjects) should or can perform in collaboration with one or more external users (actors) of the system. Each user should provide some observable and valuable result to the actors or other stakeholders of the system. Use case diagrams are twofold - they are both behavior diagrams, as the describe the behavior of the system, and also structure diagrams - as a special case of class diagrams where classifiers are restricted to be either actors or use cases related to each other with association.*

*Use case diagrams are used to specify:*

* *External requirements, required usage of a system under design or analysis (subject) - what the system is supposed to do.*
* *The functionality offered by the subject - what the system can do.*
* *Requirements the specified subject poses on its environment - by defining how environment should interact with the subject so that it will be able perform its services.*

*Below is a detailed study of the use case diagram:*

***Use Case description for “Perform Gesture”***



|  |  |
| --- | --- |
| Use Case | Track Hands |
| Primary Actor | User |
| Goal in Context | Allows the user to track finger movements |
| Preconditions | User performs hand gestures and video processing is active |
| Trigger | On tracking user could perform gestures |
| Scenario | User performs gestures |
| Priority | Essential for displaying the meaning of gestures |
| Secondary Actor | Computer |
| Exception | Video feed cannot be tracked |

***Use Case description for “Capture Video”***



|  |  |
| --- | --- |
| Use Case | Video Processing |
| Primary Actor | Web Cam |
| Goal in Context | Webcam captures live feed |
| Preconditions | User must have a webcam |
| Trigger | On capturing live feed, track movements |
| Scenario | Live feed is captured and operations are performed |
| Priority | Every frame captured should be sent for processing |
| Secondary Actor | User |
| Exception | The software breaks |

***Use Case description for “Process Video”***



|  |  |
| --- | --- |
| Use Case | Process on Received Video |
| Primary Actor | Webcam |
| Goal in Context | Operations are performed on live feed |
| Preconditions | User must have a webcam |
| Trigger | On gesturing, tracking begins |
| Scenario | Webcam captures live video which is converted into images. Operations are performed to enhance related features and gesture movements |
| Priority | Every frame must be captured |
| Exception | The software breaks |

***Use Case description for “Display Gesture Meaning”***



|  |  |
| --- | --- |
| Use Case | Display gesture meaning |
| Primary Actor | Computer |
| Goal in Context | Display the meaning according to tracking and gesture recognition |
| Preconditions | User must have a webcam |
| Trigger | The sign is recognized |
| Scenario | Enhanced images are passed to the network for recognition |
| Priority | Every gesture must be recognized |
| Secondary Actor | User, webcam |
| Exception | Gesture cannot be recognized by the software |

***Use Case description for “Track Gesture”***



***References:***

* [*https://senior.ceng.metu.edu.tr/2012/dbug/documents/SRS.pdf*](https://senior.ceng.metu.edu.tr/2012/dbug/documents/SRS.pdf)
* [*https://towardsdatascience.com/sign-language-recognition-using-deep-learning-6549268c60bd*](https://towardsdatascience.com/sign-language-recognition-using-deep-learning-6549268c60bd)
* [*https://www.ripublication.com/ijaer18/ijaerv13n9\_90.pdf*](https://www.ripublication.com/ijaer18/ijaerv13n9_90.pdf)
* [*https://github-wiki-see.page/m/CankayaUniversity/ceng-407-408-2020-2021-Sign-Language-Recognition-with-Machine-Learning/wiki/Software-Requirements-Specification-(SRS)*](https://github-wiki-see.page/m/CankayaUniversity/ceng-407-408-2020-2021-Sign-Language-Recognition-with-Machine-Learning/wiki/Software-Requirements-Specification-(SRS))