**SERENITY(Sign To Text Converter using computer vision)**

Submitted in partial fulfilment of requirements of the degree

**BACHELOR OF ENGINEERING COMPUTER ENGINEERING**

By

|  |  |
| --- | --- |
| Name | Roll Number |
| Novi Andrades | 03 |
| Alicia Dsouza | 19 |
| Alita Fernandes | 21 |
| Sharan Shetty | 60 |

Supervior

Prof. Kavita Jain

(Assistant Professor, Department of Computer Engineering)



Department of Computer Engineering,

Xavier Institute of Engineering,

Mahim, Mumbai - 410 016,

University of Mumbai (AY 2022 - 23)

**Certificate**

This is to certify that the Mini Project entitled **“SERENITY(Sign To Text Converter using computer vision )”** is a bonafide work of Novi Andrades (03) Alicia Dsouza (19) Alita Fernandes (21) Sharan Shetty(60) submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Bachelor of Engineering”** in **“Computer Engineering”**.

Prof. Kavita Jain

Supervisor

Dr. Kunal Meher Dr. Y. D. Venkatesh

Head of Department Principal

**Mini Project Approval**

This Mini Project entitled **“SERENITY(Sign To Text Converter using computer vision”** by **Novi Andrades(03), Alicia Dsouza(19), Alita Fernandes(21) and Sharan Shetty(60)** is approved for the degree of **Bachelor of Engineering in Computer Engineering**.

Examiners

1. Prof. Kavita Jain

(Internal Examiner Name and Sign)

1. **. . . . . . . . . . . . . . . . . .**

(External Examiner Name and Sign)

Date:

Place: Mahim, Mumbai

**Abstract**

Sign language serves as a communication modality reliant on diverse hand movements and actions to convey messages. The interpretation of these intricate motions presents a formidable challenge in pattern recognition. Human communication encompasses a broad spectrum of gestures and behaviors employed to convey thoughts and intentions. This study introduces a human-computer interface system tailored to the recognition of American Sign Language (ASL) gestures, translating them into corresponding textual representations. The primary aim is to mitigate the communication divide between ASL users and those reliant on written or spoken language. The proposed system leverages advanced neural network architectures, specifically Convolutional Neural Networks (CNNs) to proficiently discern and comprehend ASL gestures. This fusion of cutting-edge technologies holds the potential to significantly ameliorate accessibility and communication for ASL users, ultimately bridging the existing gap in an effective and meaningful manner.

**Contents**

1. [Introduction 11](#_TOC_250014)
   1. [Problem Definition: 11](#_TOC_250013)
   2. [Aims and Objective: 12](#_TOC_250012)
   3. [Scope of the project: 12](#_TOC_250011)
   4. [Existing System: 13](#_TOC_250010)
2. [Review Of Literature 14](#_TOC_250009)
3. [Description 16](#_TOC_250008)
   1. [Analysis 16](#_TOC_250007)
   2. [Design 17](#_TOC_250006)
   3. [Implementation Methodology 23](#_TOC_250005)
   4. [Details of Hardware and Software 23](#_TOC_250004)
   5. [Result 25](#_TOC_250003)
   6. [Sample Code 26](#_TOC_250002)
4. [Conclusion 27](#_TOC_250001)
5. References 27
   1. [Acknowledgement 28](#_TOC_250000)

### List of figures

* Figure 1: Use Case Diagram
* Figure 2: E-R Diagram
* Figure 3: DFD Level 0
* Figure 4: DFD Level 1
* Figure 5: GUI Design
* Figure 6: Block Diagram
* Figure 7: Flowchart
* Figure 8: GUI Design

### List of Tables

**Abbreviation**

**ASL - American Sign Language:** American Sign Language is a natural sign language used predominantly in the United States and parts of Canada. It is a complete, visual-gestural language with its own grammar and syntax, used by the Deaf and Hard of Hearing community for communication.

**SLT - Sign Language to Text:** Sign Language to Text refers to the process of converting sign language gestures into written or spoken language. It’s the core function of a Sign-to-Text Converter, where sign language is translated into a textual or auditory form.

**UI - User Interface:** The User Interface (UI) is the point of inter- action between a user and a system, including software applications. In the context of a Sign-to-Text Converter, the UI refers to the design and components that allow users to interact with the system, such as buttons, screens, and input methods.

**NLP - Natural Language Processing:** Natural Language Processing (NLP) is a field of artificial intelligence (AI) that focuses on the interaction between computers and human language. In the context of Sign-to-Text Converters, NLP techniques may be used to understand, interpret, and generate natural language text based on sign language input.

**API - Application Programming Interface:** An Application Programming Interface (API) defines the methods and protocols for different software components to interact with each other. In Sign-to-Text Converters, an API may enable developers to integrate the converter into various applications or services, allowing them to use sign language recognition and translation capabilities.

**Notation**

**SL - Sign Language:** Sign Language (SL) is a visual-gestural language used by the Deaf and Hard of Hearing community to communicate. It involves a combination of handshapes, movements, facial expressions, and body postures to convey linguistic information. Different regions have their own sign languages, such as American Sign Language (ASL) and British Sign Language (BSL).

**T - Text:** ”T” represents written or textual language, which is used in most forms of communication. In the context of SLT (Sign Language to Text) conversion, ”T” indicates the conversion of sign language into written text.

**SLT - Sign Language to Text:** SLT stands for ”Sign Language to Text,” which is the process of converting sign language gestures into written or textual form. This conversion allows individuals who use sign language to communicate with those who rely on written language.

**GUI - Graphical User Interface:** A Graphical User Interface (GUI) is the visual element of a software application that allows users to interact with the program using graphical elements such as windows, icons, buttons, and menus. In the context of sign language technology, a GUI may refer to the user interface of a Sign-to-Text Converter application.

**BLEU - Bilingual Evaluation Understudy:** BLEU is a metric used for evaluating the quality of machine-generated text, particularly in machine translation tasks. It measures the similarity between machine-generated text and human-generated reference text. BLEU scores help assess the accuracy and fluency of translations, including those from sign language to text

**Nomenclature**

**Gloss:** A ”gloss” refers to a specific sign or hand movement that represents a word or concept. Sign languages use a rich vocabulary of unique signs or gestures to convey words and ideas. Each gloss is a fundamental building block in sign language, where the sign’s form and motion correspond to a spoken word or concept. For example, the gloss for ”hello” in American Sign Language (ASL) involves a specific handshape and motion.

**Fingerspelling:** Fingerspelling is the act of spelling out words letter by letter using sign language. In sign languages like ASL, there are specific signs for individual letters of the alphabet. Fingerspelling is often used to convey names, places, or words that don’t have unique signs. It involves manually forming the letters with handshapes and movements. For example, when communicating a person’s name that doesn’t have a standard sign, fingerspelling is used to spell out each letter of the name.

**Interpretation:** Interpretation is the process of translating sign language into written or spoken language and vice versa. It involves skilled individuals who can bridge the communication gap between Deaf or Hard of Hearing individuals who use sign language and those who communicate in spoken or written language. Professional interpreters are trained to accurately convey the meaning, tone, and context of the conversation between both languages.

**Facial Expressions:** In sign language, facial expressions play a crucial role in conveying tone, emotion, and grammatical information. These non-manual signals are used to add nuance and context to the signs. For instance, a neutral sign for ”thank you” can be made more sincere with a warm and appreciative facial expression. Similarly, facial expressions are used to convey questions, surprise, excitement, and other emotional and grammatical nuances in sign languages.

**Handshapes:** Handshapes refer to the different configurations of the hand in sign language signs. Sign languages have a variety of hand- shapes that are combined with movements, locations, and palm orientations to create distinct signs. Each handshape represents a particular letter, word, or concept. For example, a closed fist or an open palm may represent different elements in a sign language.

1. **Introduction**

Sign language is a crucial means of communication for individuals who are deaf or hard of hearing. Instead of spoken words, it relies on hand shapes, movements, and facial expressions to convey messages. Deaf individuals often rely on sign language interpreters for everyday communication needs. However, finding skilled interpreters can be both time-consuming and costly. Sign language translation is the primary way for deaf or hard of hearing individuals to communicate effectively. Those who struggle to access reliable interpreters face significant challenges in their daily lives. Our concept is to create a system that facilitates interactions through sign language. Sign communication involves using hand movements, positions relative to the head, and facial expressions to convey meaning. To make this system work effectively, it needs to recognize various hand orientations, gestures, facial expressions, and hand positions. I propose a simple yet adaptable system capable of identifying both static and dynamic American Sign Language (ASL) motions, with a focus on the letters a to z. This focus is chosen because many individuals with hearing disabilities use ASL as their primary means of communication.

* 1. **Problem Definition:**

Effective communication between the Deaf and Hard of Hearing individuals and the general population is often hindered by the language barrier created by sign languages, which have distinct structures and are not readily understood by those who do not use them. While sign language is a vital mode of communication for the Deaf and Hard of Hearing community, there is a pressing need for a robust and efficient sign-to-text conversion system that can automatically translate sign language gestures into textual information. This system must account for the linguistic diversity among different sign languages, such as American Sign Language (ASL) and others, while providing a user-friendly and accurate means of communication. The development of a comprehensive solution that addresses these challenges and ensures effective sign-to-text conversion is critical to enhance the quality of life, education, and employment opportunities for Deaf and Hard of Hearing individuals, making it an essential area of research for the advancement of inclusive technology.

* 1. **Aims and Objective:**

Our goal is to create a technology that can bridge the linguistic diversity of sign languages, including American Sign Language (ASL) and others, by ensuring the system’s adaptability to multiple sign languages. Further- more, we aim to design a user-friendly interface that enhances accessibility and inclusivity for individuals with hearing and speech disabilities, making communication more seamless and intuitive. To achieve these aims, we will leverage advanced computer vision to improve the accuracy and reliability of sign language recognition. Our ultimate objective is to contribute to the advancement of assistive technology, providing a solution that can be employed effectively in educational, professional, and everyday communication contexts, thereby enhancing the quality of life for individuals with hearing and speech disabilities. To fulfill these aims, our specific objectives include conducting a comprehensive review of existing literature, developing a robust recognition model, implementing algorithms, creating a diverse sign language gesture database, designing and evaluating a user-friendly interface, investigating environmental factors that may affect system performance, collaborating with the Deaf and Hard of Hearing community for user feedback, and comparing the performance of our system with existing solutions. Our aim is to contribute to the dissemination of knowledge and the practical implementation of the Sign-to-Text Converter system in diverse domains, thereby enhancing accessibility and inclusivity for individuals with hearing and speech disabilities.

* 1. **Scope of the project:**
* **Gesture Recognition:**

Developing a real-time gesture recognition system that can accurately detect and interpret sign language gestures.

* **Sign Language Dictionary:**

Creating and maintaining a comprehensive sign language dictionary mapping recognized gestures to their corresponding textual representations.

* **User Interface:**

Designing a user-friendly and accessible interface for capturing video input from a camera and displaying translated text.

* **Accessibility:**

Ensuring the system’s accessibility to individuals with disabilities, including compatibility with screen readers.

* **Privacy and Security:**

Implementing robust privacy and security measures to protect users’ personal information.

* **Testing and Validation:**

Rigorously testing and evaluating the system’s accuracy and usability across a wide range of sign language gestures and users.

* **Deployment:**

Deploying the system on suitable platforms, such as mobile devices or desktop computers, to make it widely accessible.

* 1. **Existing System:**

The existing system for sign language to text translation typically relies on sign language interpreters or manual transcription. These interpreters assist speech-impaired individuals during various interactions, such as medical appointments, legal proceedings, educational sessions, and more. The limitations of the existing system include the need for human interpreters, potential communication barriers, and the unavailability of interpreters in all situations. The proposed sign language to text translator project seeks to address these limitations by providing an automated and real-time solution to bridge the communication gap between speech-impaired individuals and the general population.

1. **Review Of Literature**

In a previous comprehensive review paper, we analyzed real-time Sign Language recognition using Computer Vision. We examined preprocessing techniques, methodologies, and accuracy evaluation. A key challenge is accurate fingerspelling detection, addressed with hand segmentation, background removal, and preprocessing. Factors affecting recognition include background color, wrist angle, and camera quality. In summary, our paper offered an extensive overview of Sign Language recognition techniques, highlighting preprocessing, training methods, and challenges like background color, wrist angle, and camera quality.

[4] The proposed system is an interactive application program developed using LABVIEW software and incorporated into a mobile phone. The sign language gesture images are acquired using the inbuilt camera of the mobile phone; vision analysis functions are performed in the operating system and provide speech output through the inbuilt audio device thereby minimizing hardware requirements and expense. (Y. Madhuri, G. Anitha. and M. Anburajan., ”Vision-based sign language translation device,” 2013 International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India, 2013, pp. 565-568, doi: 10.1109/ICI- CES.2013.6508395.)

[5] In the proposed approach, the main focus is on the classification and recognition of the Indian sign language given by the dumb-deaf user in real time. Thus, the speed and simplicity of the algorithm is important. The system approach involves segmenting the hand based on the skin colour statistics, then convert that segmented image into binary, apply feature ex- traction on the binary image, for extraction of the features the techniques used are distance transformation, Discrete Fourier Transform, Probability distribution property. (Rokade, Yogeshwar Jadav, Prashant. (2017). Indian Sign Language Recognition System. International Journal of Engineering and Technology. 9. 189-196. 10.21817/ijet/2017/v9i3/170903S030.)

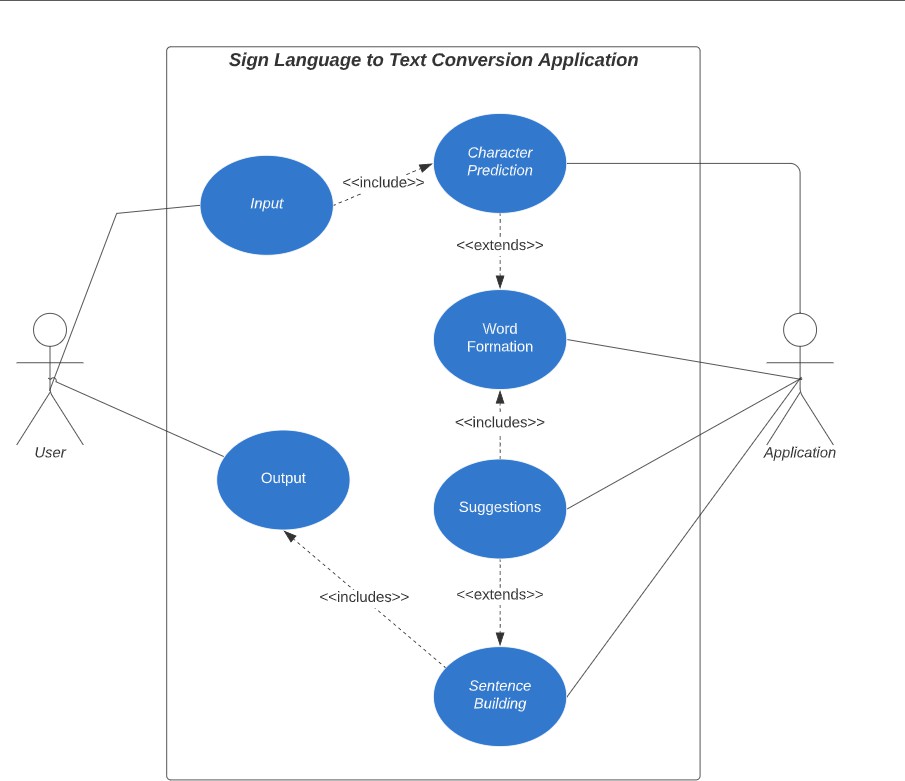
[6] Present a real-time hand tracking solution for generating a skeleton. It recognizes the palm and locates multiple landmarks on the hand. This whole is implemented using the MediaPipe framework. It achieves real time-efficient and increased performance to use in many real-time applications (Slabaugh, G., Boyes, R. and Yang, X., 2010. Multicore image processing with OpenMP [applications corner]. IEEE Signal Processing Magazine, 27(2), pp.134-138)

[6] This project is about converting the hand gesture of sign language to voice or text using Machine Learning Techniques and vice versa. In this we are going to capture a real time translation of Indian sign language using single and double hand gestures and recognize the words and convert it into text and then to speech. If the person gives speech as input it is first converted to text and then it displays the suitable sign as output and vice versa. Slabaugh, G., Boyes, R. and Yang, X., 2010. Multicore im- age processing with OpenMP [applications corner]. IEEE Signal Processing Magazine, 27(2), pp.134-138

1. **Description**
   1. **Analysis**

* **USE CASE DIAGRAM:**

The following diagram depicts the use case for the sign-to-text converter:



**Figure 1.** Use Case Diagram

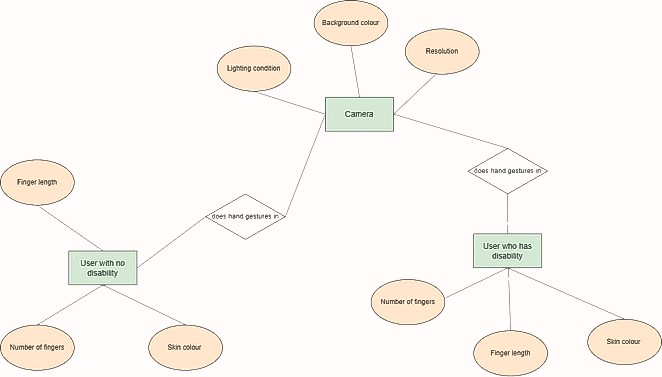
* **Feasibility Study:**

This study aims to determine whether it is technically, financially, and ethically feasible to create a sign language to text conversion technology.

* Project Scope and Objectives: Define the project’s goals, scope, and objectives. Determine the specific sign languages to be supported and the intended user base.
* Market Research: Analyze the demand for sign language to text con- version technology. Assess the size of the target market, including the deaf and hard of hearing population. Identify potential competitors and existing solutions.
* Technical Feasibility: Hardware and software requirements for gesture recognition. Availability of sign language datasets. Integration with existing platforms or devices.
* User Acceptance: Conduct surveys or interviews to understand user expectations and preferences. Consider user feedback in the design and development process.
* Report and Presentation: Compile the feasibility study into a detailed report and deliver a presentation to stakeholders, including potential investors, partners, and relevant organizations.

## Design

* **E-R Diagram:** The following diagram depicts the E-R Diagram for the sign-to-text converter:



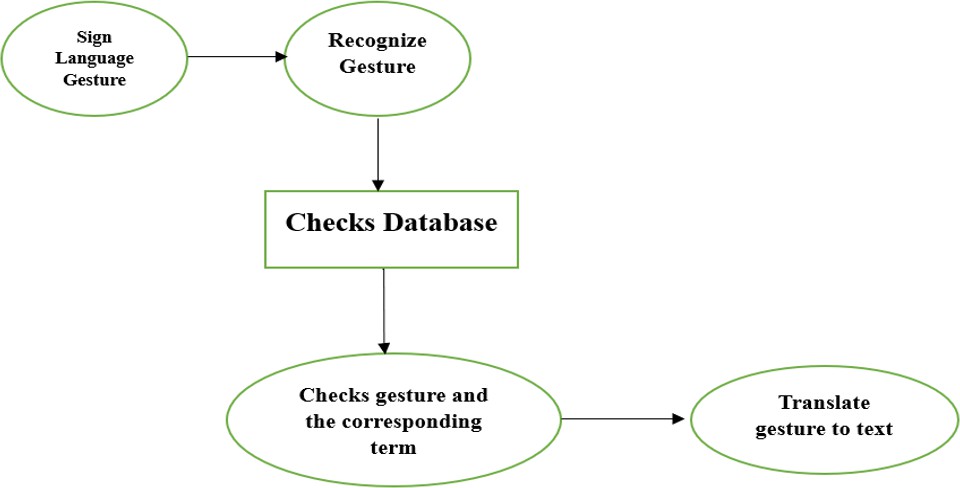
**Figure 2.** E-R Diagram

* + 1. **DFD :** The following diagram depicts the Data Flow Diagram for the sign-to-text converter:
       1. DFD LEVEL 0:



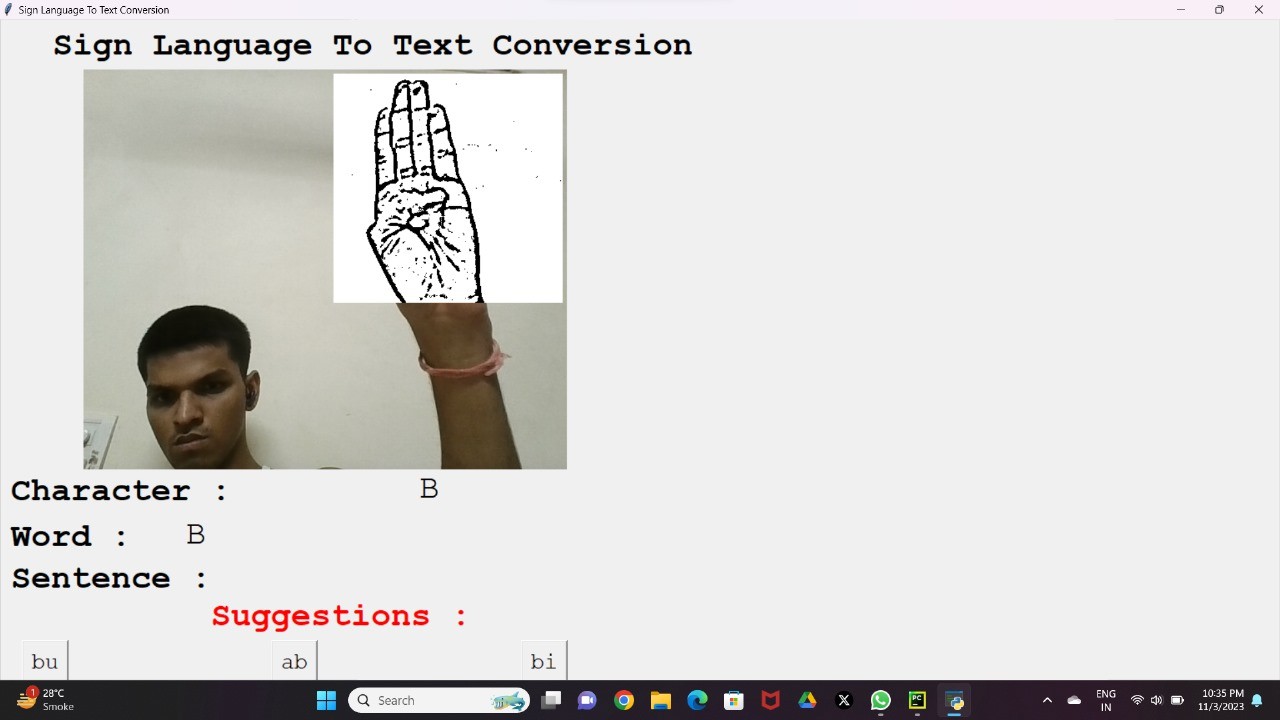
**Figure 3.** DFD Level 0

* + - 1. DFD LEVEL 1:



**Figure 4.** DFD Level 1

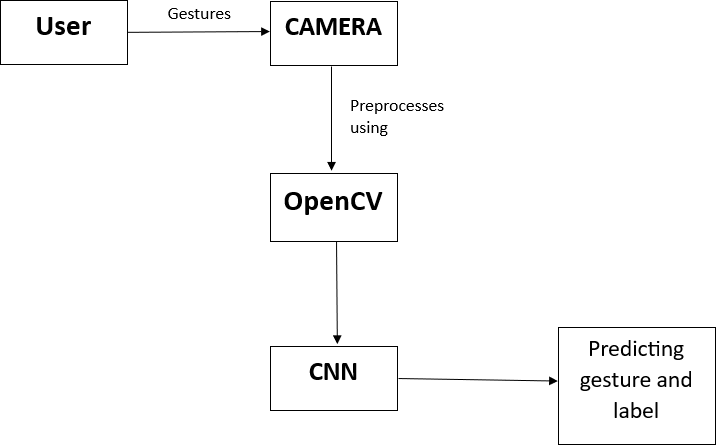
### GUI Design:



**Figure 5.** GUI Design

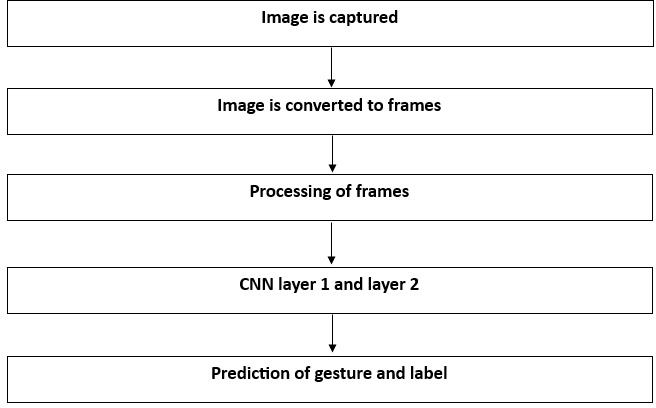
### Block Diagram:

The following diagram depicts the block diagram for the sign-to-text converter:



**Figure 6.** Block Diagram

### Flowchart:



**Figure 7.** Flowchart

## Implementation Methodology

## Details of Hardware and Software

### Software Requirements

* **Python 3.6.6:**

Python serves as the project’s primary programming language, en- abling the development of the converter’s components.

Python’s versatility and library integration capabilities make it ideal for the project.

* **Tensorflow 1.11.0:**

TensorFlow is essential for training and deploying neural network mod- els for sign language recognition. Version 1.11.0 provides key machine learning functionalities for the project.

* **OpenCV 3.4.3.18:**

OpenCV plays a central role in capturing and processing sign language gestures from video input. It offers vital computer vision capabilities for gesture recognition.

* **NumPy 1.15.3:**

NumPy is vital for efficient data manipulation and processing related to sign language recognition.

It ensures effective handling and transformation of data.

* **Matplotlib 3.0.0:**

Matplotlib aids in visualizing and presenting sign language recognition results.

It helps assess the system’s performance and data representation.

* **Hunspell 2.0.2:**

Hunspell is used for spell-checking, ensuring the correctness of the converted text. It enhances the accuracy of sign-to-text conversion.

* **Keras 2.2.1:**

Keras simplifies the development and training of deep learning models for gesture recognition. It streamlines the integration of neural networks into the project.

* **PIL 5.3.0:**

PIL/Pillow is essential for image preprocessing and preparation for gesture recognition. It aids in image manipulation and processing tasks.

**Hardware Requirements:**

* **Processor (CPU):**

Modern dual-core processor for basic functionality. Quad-core or higher for complex real-time processing.

* **Memory (RAM):**

Minimum 4 GB for basic functionality.

8 GB or more for smoother performance and larger data sets.

* **Storage:**

SSD recommended for faster access. 256 GB SSD as a starting point.

* **Graphics Processing Unit (GPU):**

Optional for basic functionality. Useful for image or video processing.

* **Operating System:**

Compatibility with the target OS (Windows, macOS, Android, etc.).

* **Input Devices:**

Camera for sign language input. Keyboard or microphone for text input.

* **Network Connectivity:**

Required for cloud features and updates.

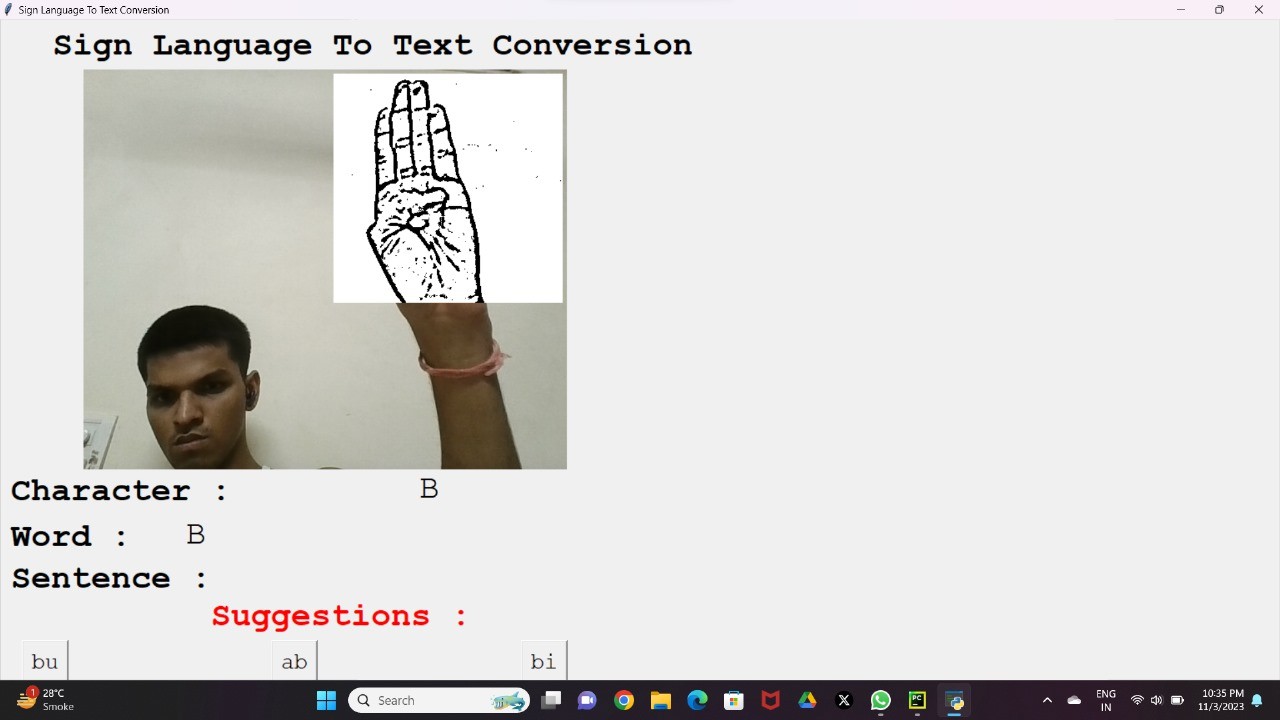
* **Display:**

Standard monitor or touchscreen display.

## Result



**Figure 8.1** GUI Design



**Figure 8.2** GUI Design

## Sample Code

class Application:

def init(self):

self.hs = SpellChecker()

self.vs = cv2.VideoCapture(0)

self.current\_image = None

self.current\_image2 = None

self.json\_file = open("Models\model\_new.json", "r")

self.model\_json = self.json\_file.read()

self.json\_file.close()

self.loaded\_model = model\_from\_json(self.model\_json)

self.loaded\_model.load\_weights("Models\model\_new.h5")

self.json\_file\_dru = open("Models\model-bw\_dru.json", "r")

self.model\_json\_dru = self.json\_file\_dru.read()

self.json\_file\_dru.close()

self.loaded\_model\_dru = model\_from\_json(self.model\_json\_dru)

self.loaded\_model\_dru.load\_weights("Models\model-bw\_dru.h5")

self.json\_file\_tkdi = open("Models\model-bw\_tkdi.json", "r")

self.model\_json\_tkdi = self.json\_file\_tkdi.read()

self.json\_file\_tkdi.close()

self.loaded\_model\_tkdi = model\_from\_json(self.model\_json\_tkdi)

self.loaded\_model\_tkdi.load\_weights("Models\model-bw\_tkdi.h5")

self.json\_file\_smn = open("Models\model-bw\_smn.json", "r")

self.model\_json\_smn = self.json\_file\_smn.read()

self.json\_file\_smn.close()

self.loaded\_model\_smn = model\_from\_json(self.model\_json\_smn)

self.loaded\_model\_smn.load\_weights("Models\model-bw\_smn.h5")

self.ct = {'blank': 0}

self.blank\_flag = 0

for i in ascii\_uppercase:

self.ct[i] = 0

print("Loaded model from disk")

self.root = tk.Tk()

self.root.title("Sign Language To Text Conversion")

self.root.protocol('WM\_DELETE\_WINDOW', self.destructor)

self.root.geometry("900x900")

self.panel = tk.Label(self.root)

self.panel.place(x=100, y=10, width=580, height=580)

self.panel2 = tk.Label(self.root) # initialize image panel

self.panel2.place(x=400, y=65, width=275, height=275)

self.T = tk.Label(self.root)

self.T.place(x=60, y=5)

self.T.config(text="Sign Language To Text Conversion", font=("Courier", 30, "bold"))

self.panel3 = tk.Label(self.root) # Current Symbol

self.panel3.place(x=500, y=540)

self.T1 = tk.Label(self.root)

self.T1.place(x=10, y=540)

self.T1.config(text="Character :", font=("Courier", 30, "bold"))

self.panel4 = tk.Label(self.root) # Word

self.panel4.place(x=220, y=595)

self.T2 = tk.Label(self.root)

self.T2.place(x=10, y=595)

self.T2.config(text="Word :", font=("Courier", 30, "bold"))

self.panel5 = tk.Label(self.root) # Sentence

self.panel5.place(x=350, y=645)

self.T3 = tk.Label(self.root)

self.T3.place(x=10, y=645)

self.T3.config(text="Sentence :", font=("Courier", 30, "bold"))

self.T4 = tk.Label(self.root)

self.T4.place(x=250, y=690)

self.T4.config(text="Suggestions :", fg="red", font=("Courier", 30, "bold"))

self.bt1 = tk.Button(self.root, command=self.action1, height=0, width=0)

self.bt1.place(x=26, y=745)

self.bt2 = tk.Button(self.root, command=self.action2, height=0, width=0)

self.bt2.place(x=325, y=745)

self.bt3 = tk.Button(self.root, command=self.action3, height=0, width=0)

self.bt3.place(x=625, y=745)

self.str = ""

self.word = " "

self.current\_symbol = "Empty"

self.photo = "Empty"

self.video\_loop()

1. **Conclusion**

This report represents a significant achievement in the development of a practical, real-time, vision-based system meticulously designed for the recognition of American Sign Language (ASL) alphabets, with a focus on meeting the unique communication needs of individuals experiencing hearing and speech disabilities individuals. This notable accomplishment has been made possible through the purposeful implementation of a sophisticated dual-layered algorithmic approach, substantially improving predictive capabilities, particularly in cases involving symbols with close visual resemblances. Consequently, this system demonstrates an exceptional proficiency in recognizing nearly all ASL symbols, provided they are presented accurately, without any background interference, and under appropriate lighting conditions. This not only underscores the substantial potential of advanced vision-based technology in the realm of ASL recognition but also sets the stage for enhanced inclusivity and more effective communication for the Deaf and Hard hearing community, marking a significant advancement in the field.

1. **References**

[1] A. Sharma, S. Panda and S. Verma, ”Sign Language to

Speech Translation,” 2020 11th International Conference on Computing, Communication and Networking Technologies (IC- CCNT), Kharagpur, India, 2020, pp. 1-8, doi: 10.1109/ICC- CNT49239.2020.9225422.

[2] W. Li, H. Pu and R. Wang, ”Sign Language Recogni- tion Based on Computer Vision,” 2021 IEEE International Conference on Artificial Intelligence and Computer Applica- tions (ICAICA), Dalian, China, 2021, pp. 919-922, doi: 10.1109/ICAICA52286.2021.9498024.

[3] V. N. T. Truong, C. Yang and Q. Tran, ”A translator for American sign language to text and speech,” 2016 IEEE 5th Global Conference on Consumer Electronics, 2016, pp. 1-2, doi: 10.1109/GCCE.2016.7800427.

[4] Y. Madhuri, G. Anitha. and M. Anburajan., ”Vision-based sign lan- guage translation device,” 2013 International Conference on Infor- mation Communication and Embedded Systems (ICICES), Chen- nai, India, 2013, pp. 565-568, doi: 10.1109/ICICES.2013.6508395.

[5] Rokade, Yogeshwar Jadav, Prashant. (2017). Indian Sign Language Recognition System. International Journal of Engineering and Tech- nology. 9. 189-196. 10.21817/ijet/2017/v9i3/170903S030.

[6] Slabaugh, G., Boyes, R. and Yang, X., 2010. Multicore image pro- cessing with openmp [applications corner]. IEEE Signal Processing Magazine, 27(2), pp.134-138

* 1. **Acknowledgement**

It gives us great pleasure to express our profound gratitude to our supervisor Prof. Kavita Jain, Department of Computer Science and Engineering for her constructive academic advice and guidance, constant encouragement, valuable suggestions, and all other support throughout this project work. We would like to appreciate all those who inspired and provided extra support to help us work on this project and make it successful. We would also like to appreciate the guidance given by other supervisors as well as the panels especially in our project presentation that their constructive feedback has made us improve our presentation skills.