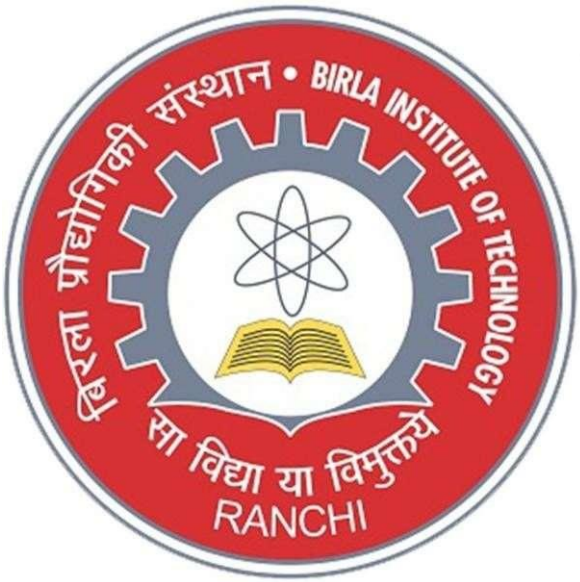
BIRLA INSTITUTE OF TECHNOLOGY
MESRA, JAIPUR CAMPUS



DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

CS 400 MINOR PROJECT

Pranav Patni
(BTECH/25002/21)
&
Namit Goyal
(BTECH/25025/21)

CASE STUDY

SIGN TO TEXT LANAGUAGE CONVERSION MODEL

PROBLEM DEFINITION

Differently abled individuals, particularly those with hearing or speech impairments, face significant challenges in communicating with people who do not know sign language. These communication barriers can affect their ability to interact in social, educational, and professional settings. The goal of the "Sign to Text Language Converter" project is to develop a web application that provides real-time translation of sign language gestures into text and vice versa, helping bridge the communication gap between sign language users and non-sign language users.

The lack of a common mode of communication between differently abled individuals using sign language and those who do not understand it leads to misunderstandings, social isolation, and difficulty in accessing services or opportunities. Existing solutions for sign language interpretation often rely on human translators, which are not always available or affordable. There is a need for an accessible, automated tool that can provide real-time, accurate sign-to-text translation and vice versa to facilitate communication.

Limited Availability of Human Translators: Human sign language interpreters are not always accessible or affordable for everyday communication needs.

Lack of Awareness: Many individuals and organizations lack awareness or knowledge of sign language, which exacerbates the communication gap.

Accuracy and Speed of Translation: Real-time communication requires fast and accurate translation of sign language gestures into text, which can be challenging due to the complexity and variations in sign language.

Diverse Sign Language Variants: There are multiple versions of sign language (e.g., American Sign Language, British Sign Language), making it important for the system to handle different variants effectively.

To solve these problems,

- To develop a web application capable of translating sign language gestures into written text in real-time.
- To ensure the system is user-friendly and accessible to both sign language users and non-users.
- To incorporate AI and machine learning models for improving translation accuracy and handling different sign language variants.
- To provide a feedback mechanism for users to correct any translation errors and improve the system's learning.

SOFTWARE REQUIREMENT SPECIFICATION

OF

SIGN TO TEXT LANAGUAGE CONVERSION MODEL

CONTENTS

AIM:

Converting the **Sign Language to Text** using an AI Model.

OBJECTIVE:

To develop a web application capable of translating sign language gestures into written text in real-time. To ensure the system is user-friendly and accessible to both sign language users and non-users. To incorporate AI and machine learning models for improving translation accuracy and handling different sign language variants.

SCOPE:

The SRS is intended to produce a software to be called the **AI-Sign Language to Text Converter System**.

- The system will focus on translating common sign language gestures into text.
- It will primarily target American Sign Language (ASL) as the first implementation, with plans to extend support to other sign languages in future versions.
- The application will be designed for use on web browsers, with potential future expansion to mobile platforms.
- The system will include basic customization options to accommodate individual preferences and regional variations in sign language.

DATA DICTIONARY:

This data dictionary forms the foundation for data management, defining how key data elements are stored, processed, and used within the "Sign to Text Language Converter" project. It ensures consistency and clarity across system development, integration, and deployment.

1. USER DATA

Field Name	Data Type	Description	Example Values
user_id	Integer	A unique identifier for each user.	1001, 1002, 1003
user_name	String	Name of the user using the system (either a sign language user or a non-sign language user).	"JohnDoe", "JaneSmith"
email	String	The email address of the user for system notifications and login.	"john@example.com"
password	String	The user's encrypted password for login.	"hashed_password_value"
user_type	Enum	Defines whether the user is a sign language user or a non-sign language user.	["Sign User", "Non-Sign User"]
preferred_language	String	The preferred language of the user for the interface and output text.	"English", "Spanish"

2. Sign Language Data

Field Name	Data Type	Description	Example Values
gesture_id	Integer	A unique identifier for each sign language gesture.	101, 102, 103
gesture_name	String	Name of the specific sign or gesture in the sign language.	"hello", "thank you"
gesture_video	Binary (File)	A video clip or image sequence that represents the sign language gesture.	video_101.mp4
gesture_type	Enum	Category of gesture (e.g., finger spelling, facial expression, hand movement).	["Finger Spelling", "Facial Expression"]
gesture_language	Enum	The specific sign language the gesture belongs to (e.g., ASL, BSL).	["ASL", "BSL"]
gesture_duration	Float	Duration of the gesture in seconds.	1.5, 2.0

3. Translation Data

Field Name	Data Type	Description	Example Values
translation_id	Integer	Unique identifier for a translation event (from sign to text or vice versa).	2001, 2002
input_type	Enum	Defines whether the input is a sign gesture or text.	["Sign", "Text"]
input_value	String/Binary	The input to be translated, either a sign gesture (as video/image) or text.	"hello", video_201.mp4
output_value	String	The translated output (if input is sign, output is text, and vice versa).	"Hello", "Gracias"
confidence_score	Float	Confidence level of the AI translation for the input gesture or text.	0.85, 0.92
translation_status	Enum	Indicaes whether the translation is successful, pending, or need revision.	["Success", "Pending", "Failed"]

4. System Performance Data

Field Name	Data Type	Description	Example Values
request_id	Integer	Unique identifier for each system request (such as translation or feedback submission).	4001, 4002
response_time	Float	The time taken by the system to complete a translation request, in seconds.	2.5, 1.8
error_code	String	Error code generated in case of system failure or issue.	"ERR_101", "ERR_404"
error_message	String	Description of the error encountered during translation or system operation.	"Invalid gesture input", "Network error"

5. Session Data

Field Name	Data Type	Description	Example Values
session_id	Integer	A unique identifier for each user session in the system.	5001, 5002
user_id	Integer	The unique ID of the user associated with the session.	1001, 1002
session_start	DateTime	The timestamp when the session started.	2024-09-01T08:00:00
session_end	DateTime	The timestamp when the session ended.	2024-09-01T08:30:00
actions_performed	List	List of actions (e.g., sign-to-text translation, feedback submission) performed in the session.	["Translation", "Feedback"]

EVENT LIST:

- **User Registration**: Admin registers users by inputting details (name, ID) and optionally capturing their sign language preferences or personal data, which is stored in the system.
- **Sign Gesture Detection**: The system captures the user's sign gestures via a camera and processes the video frames for recognition.
- **Sign Gesture Recognition**: The system analyzes the captured gestures and compares them with stored gesture data to identify the corresponding sign language meaning.
- **Translation to Text Recording**: Once the sign gesture is recognized, the system translates it into text, storing the result along with a timestamp for future reference (check-in for translation events).
- **Report Generation**: Admin or user can request reports on translation history (daily, weekly, custom) based on stored data, including gesture-to-text accuracy rates.
- **Manual Translation Updates**: Admin or users can update or correct translation records in case of incorrect translations or to train the system for better accuracy.
- **Notifications**: The system generates alerts for issues like unrecognized gestures, system errors, or missing translations.
- **System Maintenance**: Admin performs regular maintenance tasks like data backups, AI model updates, and system optimization to ensure smooth translation functioning.

DATA FLOW DIAGRAM(DFD):

LEVEL 0

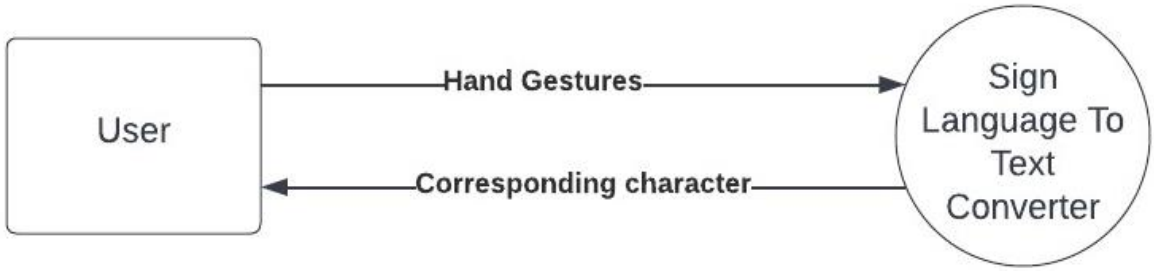


Fig.1 DFD Level 0

LEVEL 1

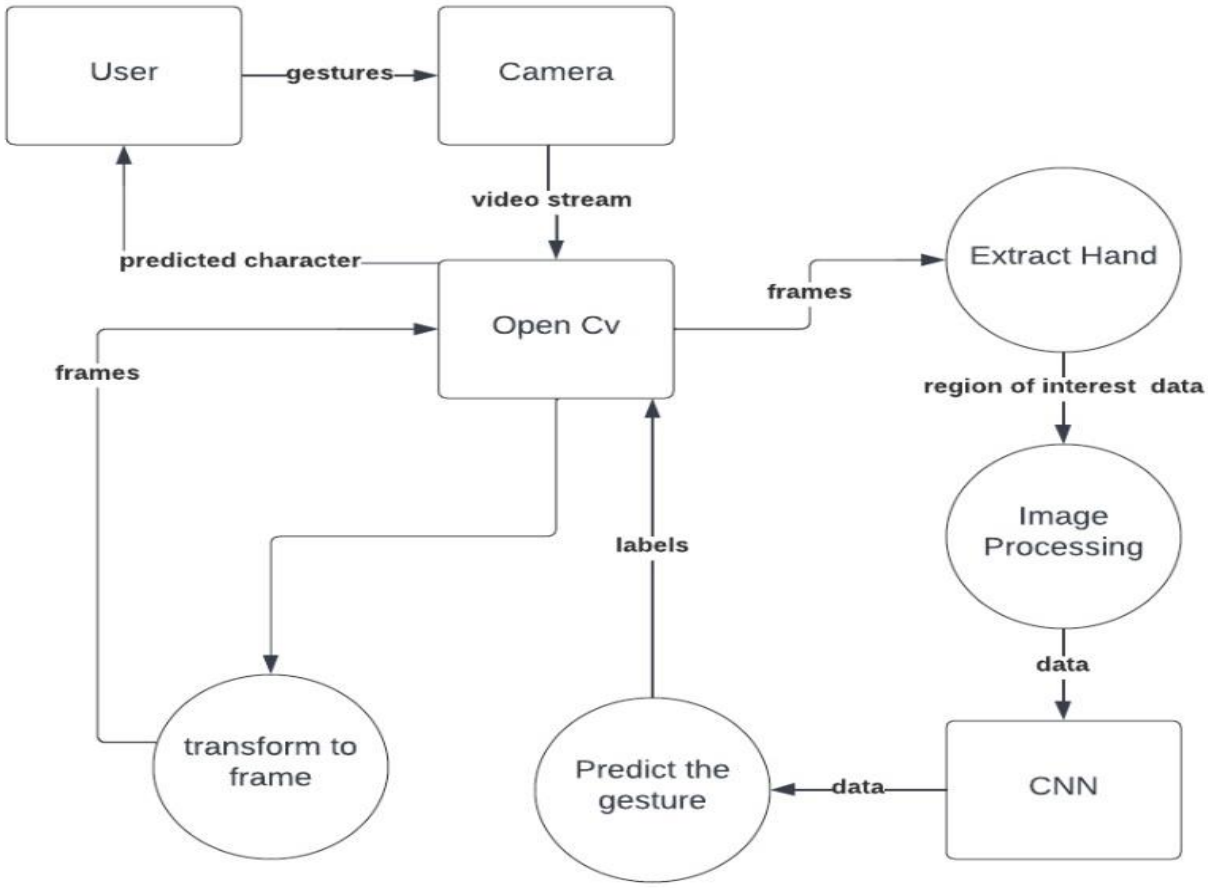


Fig.2 DFD Level 1

ENTITY RELATIONSHIP DIAGRAM (ERD):

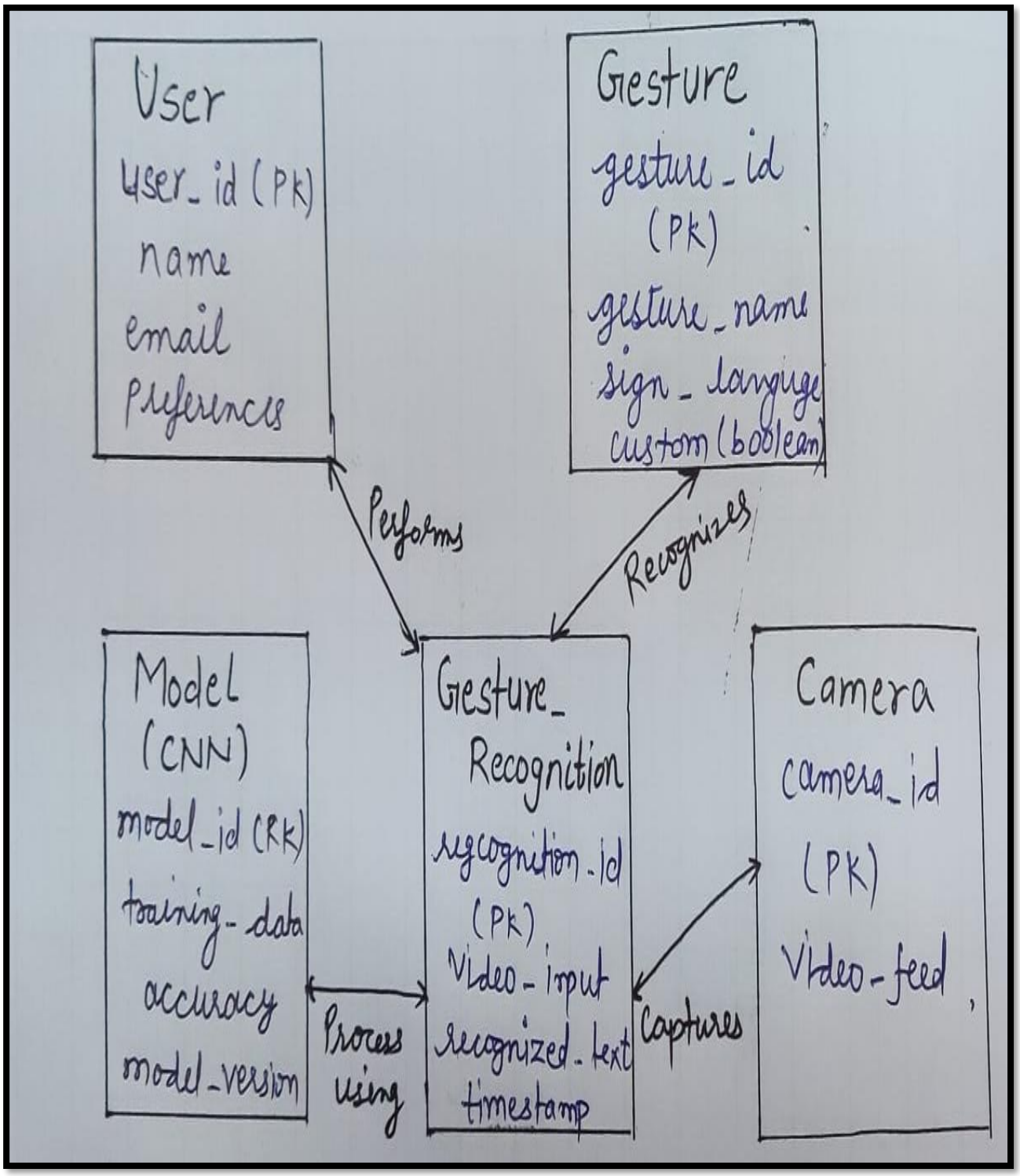


Fig.3 ER Diagram

USE CASE DIAGRAM

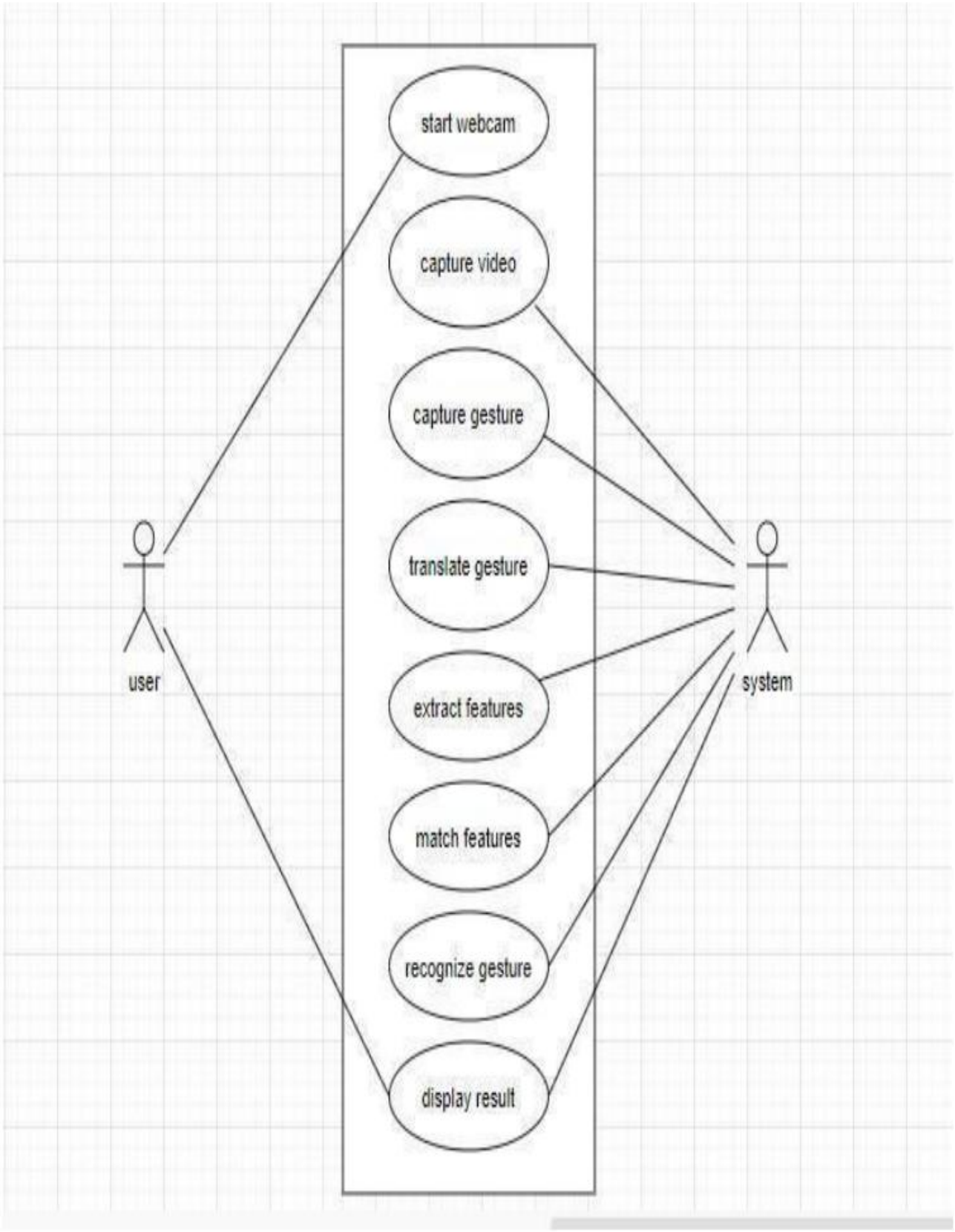


Fig.4 USE Case Diagram

MINI SPECIFICATIONS:

1. User Registration

- **Objective**: Admin registers users by inputting details such as name, email, and sign language preference.
- **Inputs**: Name, email, user type (sign language user or non-sign language user), sign language preferences.
- **Outputs**: User account is created, and data is stored in the system database.
- **Preconditions**: Admin has access to the registration panel.
- **Postconditions**: User is successfully registered and can log into the system.

---

2. Sign Gesture Detection

- **Objective**: Capture a user's sign gestures in real-time using a webcam or other camera devices.
- **Inputs**: Live video feed or uploaded video of a sign language gesture.
- **Outputs**: Extracted image frames from the video feed.
- **Preconditions**: The camera is enabled and functional; the user is in view of the camera.
- **Postconditions**: Video is processed for gesture recognition.

---

3. Sign Gesture Recognition

- **Objective**: Analyse the captured video frames and recognize the sign language gestures using AI/ML models.
- **Inputs**: Processed image frames from the video feed.
- **Outputs**: Identified sign language gesture along with confidence score.
- **Preconditions**: The sign gesture detection process is completed, and AI/ML models are trained.
- **Postconditions**: Gesture is successfully recognized, or an error is flagged if unrecognized.

---

4. Translation to Text Recording

- **Objective**: Convert the recognized sign gesture into corresponding text and store the translation with a timestamp.
- **Inputs**: Recognized sign gesture.
- **Outputs**: Translated text, timestamp, and event log entry.
- **Preconditions**: Successful gesture recognition.
- **Postconditions**: Translation is displayed to the user and logged for future reference.

---

5. Report Generation

- **Objective**: Generate translation reports based on user interactions, including translation history and accuracy rates.
- **Inputs**: Admin or user request for report generation, including date range or custom filters.
- **Outputs**: Detailed reports on translation history (daily, weekly, or custom).
- **Preconditions**: Sufficient data must be available in the system database for report generation.
- **Postconditions**: Reports are generated in the requested format (PDF, CSV, etc.).

---

6. Manual Translation Updates

- **Objective**: Allow admin or user to manually update or correct translation entries in case of errors.
- **Inputs**: Translation ID, updated or corrected text, optional feedback.
- **Outputs**: Updated translation records in the database.
- **Preconditions**: A translation entry exists, and the user or admin has access rights to modify it.
- **Postconditions**: Translation is updated, and the system logs the change.

---

7. Notifications

- **Objective**: Generate system alerts for various issues like unrecognized gestures, errors, or missing translations.
- **Inputs**: System status, unrecognized gestures, errors in recognition or translation.
- **Outputs**: Notification messages (email, UI alerts).
- **Preconditions**: The system encounters an error or predefined trigger event.
- **Postconditions**: User or admin receives a notification with details on the issue.

---

8. System Maintenance

- **Objective**: Allow admin to perform regular system maintenance tasks like data backups and AI model updates.
- **Inputs**: Admin-initiated maintenance requests (backup, update, etc.).
- **Outputs**: System status report, updated AI models, or backup confirmation.
- **Preconditions**: Admin has access rights, and the system is in a stable state for maintenance.
- **Postconditions**: System continues functioning smoothly after maintenance completion.

LIMITATIONS:

1.Accuracy of Gesture Recognition

- Description: The accuracy of gesture recognition depends heavily on the quality of the video input and the AI/ML model's training.
- Limitation: Inconsistent lighting, camera angles, or background noise can reduce the system's ability to accurately detect and recognize gestures.
- Impact: Misinterpretation of gestures may result in incorrect translations, reducing the effectiveness of the system.

---

2. Limited Sign Language Variants

- Description: The system may initially support only a few sign languages (e.g., ASL or BSL), while there are many regional sign languages.
- Limitation: Users from different regions or countries may use sign languages that are not supported by the system.
- Impact: The system may not be usable for people who use less common sign languages, limiting the audience.

---

3. Real-time Performance

- Description: The project involves real-time gesture detection, processing, and translation.
- Limitation: Delays in processing or poor system performance can slow down the real-time translation, leading to lag in communication.
- Impact: Delays may hinder the natural flow of conversations and reduce user satisfaction.

---

4. Complex Gesture Recognition

- Description: Some sign language gestures are highly nuanced and involve facial expressions, body movements, and context.
- Limitation: The system may struggle to recognize and accurately translate complex or multi-part gestures that depend on context.
- Impact: Users may need to simplify their gestures, leading to less precise communication or limited expression.

---

5. Limited Vocabulary and Contextual Understanding

- Description: Sign languages rely on context and can have many meanings for a single gesture depending on the situation.
- Limitation: The system may have limited vocabulary or struggle to understand gestures in different contexts.
- Impact: Miscommunication or incomplete translations can occur if the system cannot interpret the gesture based on context.

---

6. Dependency on Internet Connectivity

- Description: The system may require constant internet connectivity for cloud-based processing and model updates.
- Limitation: In areas with poor or no internet connectivity, the system may not function efficiently or at all.
- Impact: Users in remote areas or with limited internet access may face challenges using the system.

CONCLUSION:

The *Sign to Text Language Converter* project aims to provide a technological solution to bridge communication gaps for individuals with hearing and speech impairments by converting sign language into text and vice versa. Through the integration of advanced technologies like computer vision, artificial intelligence, and machine learning, this system can translate sign gestures into text in real-time, fostering better communication between sign language users and non-sign language speakers.

This project addresses the crucial need for inclusivity and accessibility, providing users with a user-friendly platform to translate and communicate across different languages. While the system offers substantial benefits, it also comes with limitations such as gesture recognition accuracy, real-time performance challenges, and the need for continuous AI model updates.

In conclusion, the *Sign to Text Language Converter* is a significant step toward enhancing communication for specially-abled individuals. With ongoing development, user feedback, and improvements in AI accuracy, the system has the potential to become a widely adopted tool for improving accessibility and inclusion in everyday life, workplaces, and educational settings. Despite certain challenges, the project holds promise for making a positive social impact by empowering individuals and creating an accessible communication channel for all.

FUTURE SCOPE:

Future Scope: Text to Sign Language Conversion (Vice-Versa Model)

As part of the future scope for the *Sign to Text Language Converter*, the development of a **Text to Sign Language Conversion** model would be a natural extension to the existing system. This enhancement would enable communication in the reverse direction, translating written or spoken text into sign language gestures, providing a complete communication solution for individuals who rely on sign language.

1. Text to Sign Language (TSL) Translation Model

- **Functionality**: The system would take text input, either typed or spoken (using speech-to-text technology), and translate it into sign language. The translation would be output in the form of a video showing a virtual avatar or animated representation performing the appropriate sign gestures.
- **Use Cases**:
  - Deaf or hard-of-hearing users could receive information from non-sign language users through sign language output.
  - Educational tools for sign language learners, where users can input text and see the corresponding sign gestures.
  - Customer service or workplace environments where text-based information can be seamlessly communicated in sign language.

2. Integration of Avatars for Gesture Rendering

- **Advanced Virtual Avatars**: The system could integrate virtual avatars that mimic human gestures, including facial expressions and body movements essential to sign language.
- **3D Rendering**: For a more immersive and realistic experience, 3D avatars can be used to perform the signs in a more lifelike and interactive manner, enhancing the clarity of communication.

3. Real-time Text to Sign Language Interpretation

- **Speech Recognition**: Integrating speech recognition into the system would allow real-time translation of spoken language into sign gestures. This can be particularly useful in live meetings, classrooms, or events where quick and accurate interpretation is essential.
- **Text Streaming**: For applications such as conferences or media consumption, the system could translate streaming captions or subtitles into sign language for accessible content delivery.