OBJECTIVE

- Develop a system to detect and translate **Indian Sign Language (ISL)** into English text and speech.
- Bridge the communication gap between the **deaf community and hearing people**.

METHODOLOGY

- **Hand landmarks** from **MediaPipe** are used to track hand gestures efficiently.

- Signs are detected using **deterministic methods** for higher accuracy.

- A **desktop application** is developed to display recognized signs and generate corresponding voice output.

ALTERNATIVE MODELS

- Traditional **Deep Learning (DL) models** like **LSTM, YOLO, etc.** require large datasets of images/videos and high-powered GPUs for training.

- **MediaPipe-based approach** allows users to train their own models **without collecting thousands of images or videos**.

WHY MEDIAPIPE?

- 1.No need of huge data
- 2.No need of GPU
- 3.No upper limit on how many signs can be made
- 4.Better accuracy scores
- 5.Flexible to compose sentences

DATASET & TRAINING

- **Minimal dataset requirement**: Only **2 to 5 images/videos per sign** are sufficient for training.

- Hand trajectory tracking is performed using **NumPy values of landmarks**.

- **Motion differences** (instead of derivatives) are calculated to avoid complications due to varying speeds.

TECHNOLOGY STACK

- **MediaPipe** – For hand tracking and gesture recognition.
- **OpenCV** – For video/image processing.
- **NumPy** – For tracking changes in hand motion.
- **Text-to-Speech (TTS) API** – To generate voice output.

PROPOSED IMPLEMENTATION

- The system will first recognize the following signs:
- **Alphabets (A-Z)**
- **Digits (0-9)**
- **Common words** (e.g., Hello, Thank You, Yes, No, Please, Sorry)
- These basic signs will form the foundation for expanding to **more complex words and phrases** in the future.

STATIC SIGN RECOGNITION

- **Hand Landmarks & Orientation**
- **MediaPipe** is used to detect hand landmarks.
- Each sign is recognized based on **X, Y, Z coordinates** of hand landmarks.
- **Hand orientation** is calculated using differences in key landmark positions.

STATIC SIGN RECOGNITION (cont.)

- **Single-Hand vs. Double-Hand Signs**

- **Single-hand signs** can be inferred directly based on landmark positions.

- **Double-hand signs** require relative positioning and orientations of both hands.

STATIC SIGN RECOGNITION (cont.)

- **Data Logging for Accuracy**

- **Log functions** store landmark positions, orientations, and hand states (open/closed).
- **User-Defined Class** stores raw landmark data, hand type (left/right), and hand angles.
- Raw coordinates (0 to 1) are scaled based on screen width for better accuracy.

EXAMPLE STATIC SIGN AND IMPLEMENTATION

- **Recognized Static Signs**
- Common ISL words like **"word," "you," "me," "place," "time," "this," "that," etc.**
- Configured using **X, Y, Z coordinates** and orientations of both hands.

EXAMPLE FOR STATIC SIGN & IT'S IMPLEMENTATION (cont.)

- **Rule-Based Detection**

- **Letter ‘A’** → Thumb open, all other fingers closed.

- **Letter ‘C’** → All fingertip landmarks aligned in X-axis, thumb Y-coordinate higher.

- **Digit ‘1’** → Index finger open, all others closed.

DYNAMIC SIGN RECONITION

- **Capture Process:**

- **Video Capture:**
  - Dynamic sign sequences are recorded at **30 frames per second** using OpenCV2.
- **Full Sequence Tracking:**
  - The system tracks hand landmarks continuously from the **start to the end** of the action.

DYNAMIC SIGN RECONITION (cont.)

- **Step-by-Step Processing:**

- **Hand Landmark Tracking:**
  - Continuously monitor hand landmarks throughout the dynamic gesture.
- **Frame-by-Frame Logging:**
  - Log each frame’s hand landmarks and orientations **live** during the sign.
- **Rule Definition from Orientation:**
  - Use the captured hand orientation information to define rules for dynamic sign detection.

DYNAMIC SIGN RECONITION (cont.)

- **Data Conversion:**
- Convert the hand landmark data into **NumPy arrays** and store them as dictionaries for each timestamp.
- **Trend Analysis:**
- Compute the **differences between NumPy arrays** to capture the motion trend.
  - *Note:* Derivatives are avoided as variations in speed can skew results.
- **Threshold Application:**
- Apply a threshold to filter out minor, irrelevant movements (noise due to lighting changes or camera fluctuations).

EXAMPLE FOR DYNAMIC SIGN & IT'S IMPLEMENTATION

- **Detecting the Word “WHAT”**

- **Initial Posture:**
  - The sign begins with an **open palm facing upward**.

- **Movement Pattern:**
  - The palm is moved **right to left at least two times**.

EXAMPLE FOR DYNAMIC SIGN & IT'S IMPLEMENTATION (cont.)

- **Detecting the Word "WHAT"**

- **Axis Consideration:**
  - **X-direction:** Significant movements are observed.
  - **Y and Z directions:** Changes are negligible.

- **Rule Implementation:**
  - Check the differences in **X-coordinates**.
  - Detect alternating patterns (positive-to-negative and vice versa) at least **twice** to confirm the sign "WHAT".

CONCLUSION & FUTURE SCOPE

- **Immediate Goal:** Convert ISL to English with **fast & accurate sign detection**.

- **Future Expansion:**
- Cover **more words and phrases**.
- Enable **two-hand gesture recognition**.
- Develop a **mobile-friendly version**.

- **Final Aim:** Bridge the **communication gap** between **Deaf and Hearing individuals**.