

# HAND GESTURE RECOGNITION FOR COMMUNICATION

Using MediaPipe + Machine Learning

PRESENTED BY  
BHAVESH CHINDALIYA  
SANSKRUTI GOSAVI

# PROBLEM:

- COMMUNICATION GAP FOR PEOPLE WITH SPEECH/HEARING DIFFICULTIES.

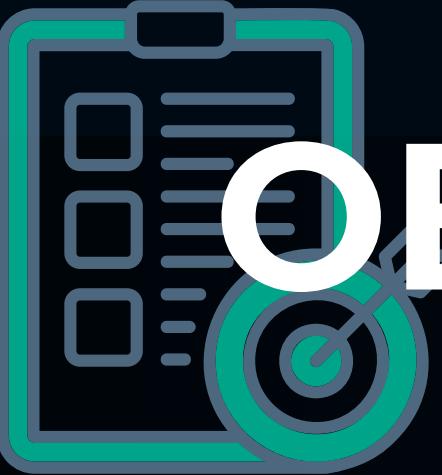
# SOLUTION:

- DETECT GESTURES VIA WEBCAM → TRANSLATE INTO WORDS.

# BENEFIT:

- EVEN NON-SIGN-LANGUAGE USERS CAN UNDERSTAND GESTURES.

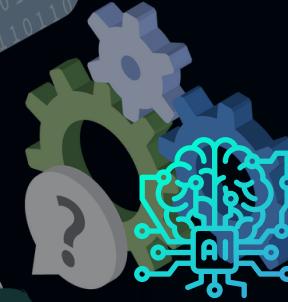




# OBJECTIVES



Build a real-time gesture detection system.



Train ML model to classify gestures.



Provide an assistive communication tool.



# TECHNOLOGY STACK

- Python (Core Programming)
- OpenCV (Webcam & Image Processing)
- MediaPipe (Hand landmark detection)
- Scikit-learn (Random Forest Model)
- CSV (Dataset handling)

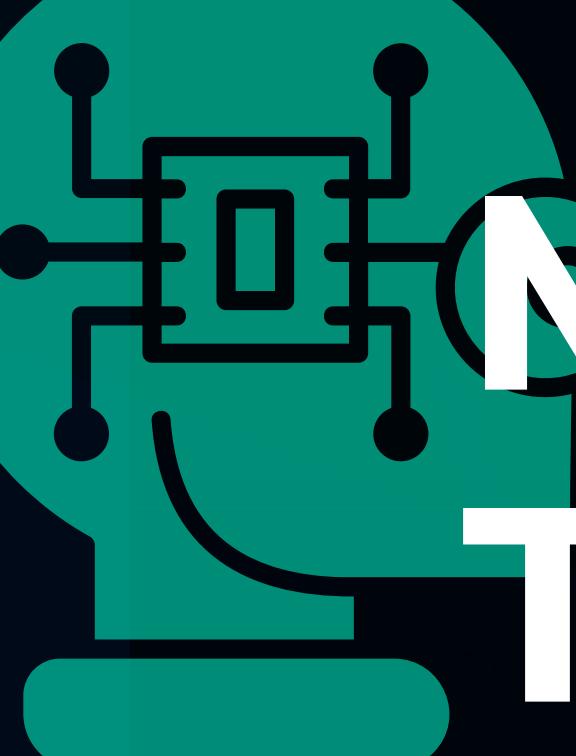
# SYSTEM ARCHITECTURE

- Webcam captures frame
- MediaPipe detects hand landmarks
- Features extracted (x,y positions)
- Model (Random Forest) predicts gesture
- Output displayed as text

# • DATASET

- Total samples : 15
- Features per sample: 63 - x, y coordinates of hand landmarks
- Number of gesture classes: 3
  - Fist
  - Thumbs Up
  - Peace
- Dataset is relatively small but covers multiple gesture types.
- Used for training + testing the classifier.





# MODEL TRAINING

- Algorithm Used: Random Forest Classifier (Scikit-learn)
- Input: 63 features (landmark coordinates) from dataset
- Output: Trained Model → gesture\_rf.pkl
- Target Classes: {Fist, Thumbs Up, Peace}
- Training Approach:
- Preprocessed dataset from MediaPipe landmarks
- Split data into train/test sets
- Trained Random Forest → saved as .pkl for real-time prediction



# OUTPUT SCREENSHOTS



# RESULTS & OBSERVATIONS

- Successfully recognized gestures in real-time.
- Works well in good lighting & clear hand visibility.
- Faster compared to manual sign language interpreters.

# CONCLUSION

- Built a real-time assistive tool.
- Bridges communication gap.
- Practical use case in healthcare & education.



# THANK YOU