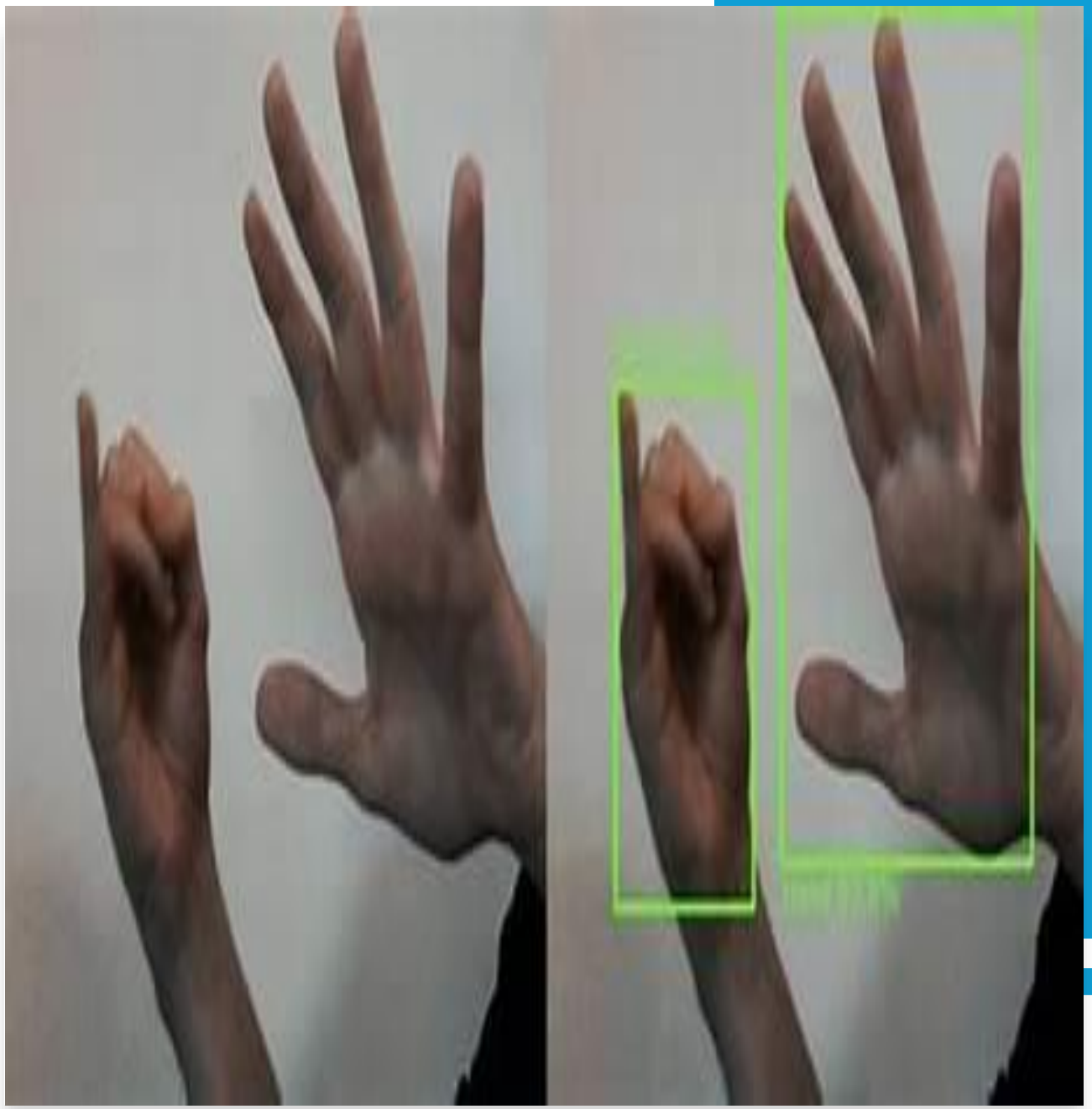
CAPSTONE PROJECT

SIGN LANGUAGE DETECTION



PRESENTED BY

STUDENT NAME:ABHISHEK NISHAD

COLLEGE NAME:ITM COLLEGE OG MANAGEMENT

DEPARTMENT:BCA

EMAIL ID:nabhishek289@gmail.com

AICTE STUDENT ID:

STU6765189215cb51734678674

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

Communication between hearing-impaired individuals and those unfamiliar with sign language presents a significant barrier in daily interactions. Traditional methods of sign language interpretation rely on human translators, which may not always be available or practical. Additionally, many public and digital services lack inclusivity for sign language users, thereby limiting accessibility and engagement for the hearing-impaired community. There is a need for a system that can bridge this communication gap by recognizing and interpreting sign language efficiently and accurately.

PROPOSED SOLUTION

- The proposed system is a real-time sign language detection model using deep learning techniques. The system captures hand gestures via a webcam, processes the images, and classifies them into corresponding alphabetic characters or words using a convolutional neural network (CNN). The model is trained on a dataset of labeled sign language gestures, enabling it to learn spatial patterns and hand positions. The detected output is displayed as text, which can be extended to speech output for broader communication use.

SYSTEM APPROACH

- **Programming Language:** Python
- **Libraries/Frameworks:** TensorFlow, Keras, OpenCV, NumPy, Matplotlib
- **Tools:** Jupyter Notebook, Google Colab (for training), Flask (for deployment), HTML/CSS (frontend interface)
- **Dataset:** American Sign Language (ASL) Alphabet Dataset (image-based)
- **Hardware:** Standard webcam for real-time input

ALGORITHM & DEPLOYMENT

Algorithm (Model Architecture):

- **Preprocessing:**
  - Image resizing (e.g., 64x64)
  - Normalization and grayscale conversion
- **Model:**
  - CNN with multiple convolutional + max-pooling layers
  - ReLU activation and dropout for regularization
  - Fully connected layers for classification
  - Softmax output layer for multi-class prediction

Training:

- Optimizer: Adam
- Loss Function: Categorical Cross-Entropy
- Epochs: 20–50
- Batch Size: 32
- Validation split: 20%

Deployment:

- The trained model is exported as .h5
- A Flask server handles video input and model prediction
- The frontend captures video frames and displays predicted output in real-time

RESULT

- Real-time webcam feed
- Detected sign gesture highlighted
- Corresponding alphabet/word displayed on-screen

CONCLUSION

The developed sign language detection system effectively recognizes and interprets hand gestures corresponding to the ASL alphabet using deep learning. The integration of a convolutional neural network enables high accuracy in real-time recognition. This system significantly enhances communication accessibility for the hearing-impaired and can serve as a bridge between sign language users and the general population.

FUTURE SCOPE

- Extend the model to recognize dynamic gestures (complete sentences, continuous signing).
- Add support for additional sign languages (e.g., BSL, ISL).
- Integrate voice output to provide speech assistance.
- Deploy the system on mobile platforms using TensorFlow Lite.
- Improve accuracy by using advanced models like ResNet, or adding hand-tracking sensors for better gesture capture.

REFERENCES

•American Sign Language Dataset: datamunge/sign-language-mnist

•Chollet, F. (2015). Keras.

•Abadi, M., et al. (2016). TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems.

•OpenCV Documentation:

•Relevant research papers on sign language recognition using deep learning (IEEE, Springer)

GitHub Link: https://github.com/Abhiabhishek65/SIGN-_LANG_DET/upload/main

Thank you