"DEEP LEARNING FOR SIGN LANGUAGE DETECTION AND CAPTION TRANSLATION"

TEAM MEMBERS:

1. HIMANI SHAH	017411407
2. KUSH BINDAL	017441359
3. MANJOT SINGH	017557462
4. SAQIB CHOWDHURY	017514978
5. SUGANDHA CHAUHAN	017506190
6. HAMSALAKSHMI RAMACHANDRAN	017423666

PROBLEM DEFINITION

01

Awareness Deficit in Sign Language Adoption

Sign language serves as an essential means of communication for people with hearing disabilities. Nonetheless, the limited awareness of sign language creates major obstacles in everyday life.

02

Challenges in Existing Gesture Recognition Technologies

This project aims to utilize action recognition methods to effectively detect sign language gestures. In contrast to conventional gesture recognition systems that mainly depend on static hand shape identification, this project will examine motion patterns to better interpret sign language.

03

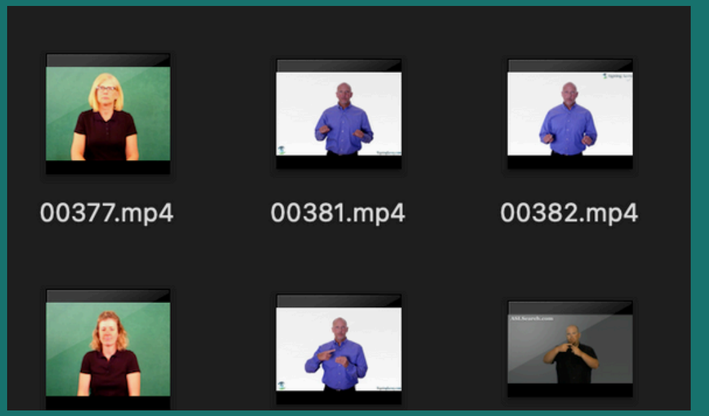
Emerging Need for Inclusive Solutions

Our approach builds a ASL recognition system that extracts hand keypoints from video clips using segmentation method, processes them with a RESNET-18 and translates the output into multiple languages via a lightweight model (Helsinki-NLP/opus-mt-en-hi).

DATASET

WLASL (Word-Level American Sign Language)

- Comprises approximately 12,000 processed video clips covering 2,000 common ASL words
- Each video captures a signer performing a specific ASL sign
- Many signs (especially frequent ones) have up to 50 different samples, captured from various users and conditions.
- Videos are typically ~1–3 seconds long.




```
{
    "gloss": "book",
    "instances": [
        {
            "bbox": [
                385,
                37,
                885,
                720
            ],
            "fps": 25,
            "frame_end": -1,
            "frame_start": 1,
            "instance_id": 0,
            "signer_id": 118,
            "source": "aslbrick",
            "split": "train",
            "url": "http://aslbricks.org/New/ASL-Videos/book.mp4",
            "variation_id": 0,
            "video_id": "69241"
        },
```


EgoHands dataset (For Segmentation finetune)

- 48 Google Glass Videos from First-Person Perspective
- There are 100 labeled frames for each of the 48 videos for a total of 4,800 frames.
- 4 Activity Scenarios: playing cards, chess, jenga and solving puzzles.
- Pixel-Level Hand Annotations and includes occlusions, motion blur, and varying skin tones.

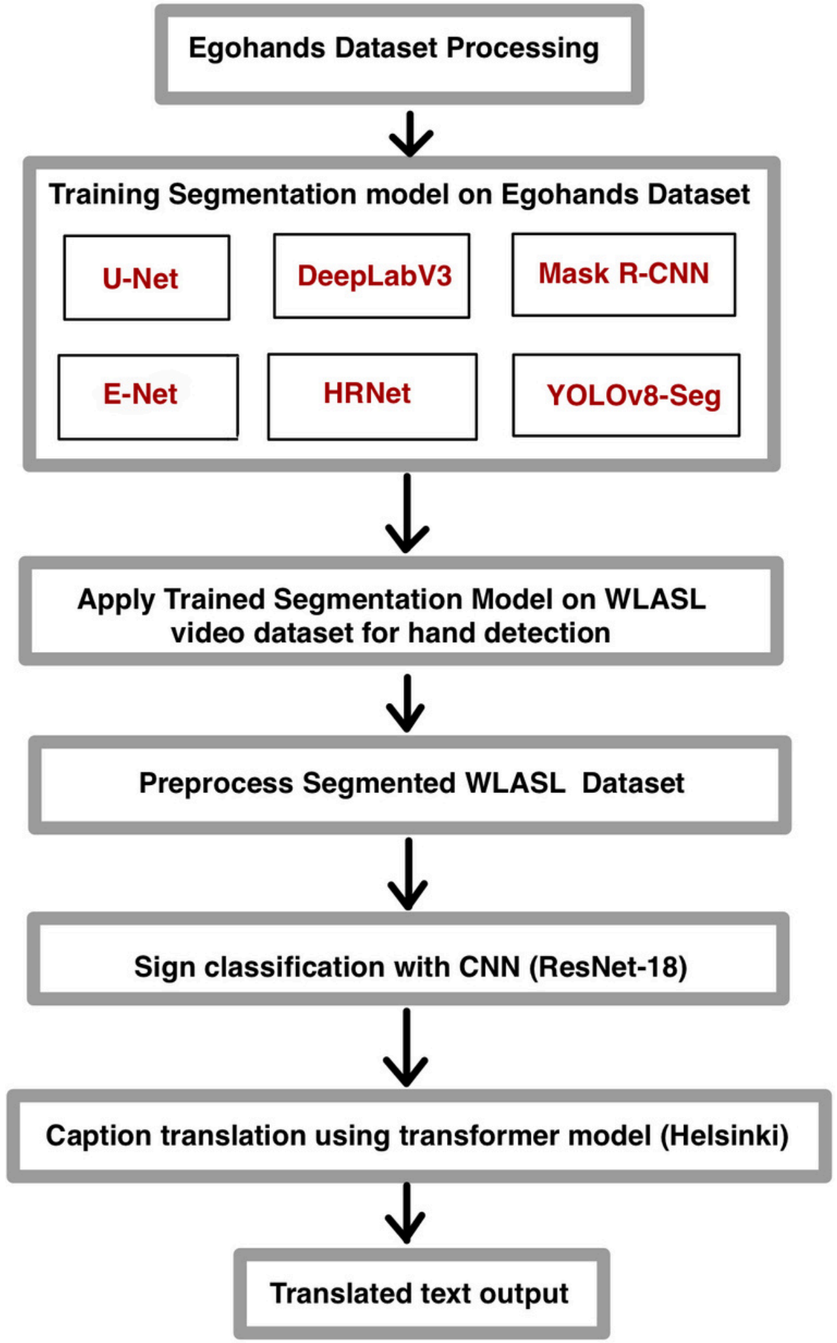
METHODOLOGY

The project begins with the EgoHands dataset, which contains hand-annotated images using polygon masks. These annotations were converted into a format suitable for segmentation training across various models.

A variety of segmentation architectures were trained using the annotated EgoHands dataset and their effectiveness is compared on WLASL dataset.

The segmented WLASL images were then used to train ResNet-18, to classify hand gestures

Based on classification performance, the best-performing segmentation model was selected. Finally, its predicted labels were passed through a Helsinki-NLP transformer for caption translation to Hindi, completing the sign-to-text pipeline.



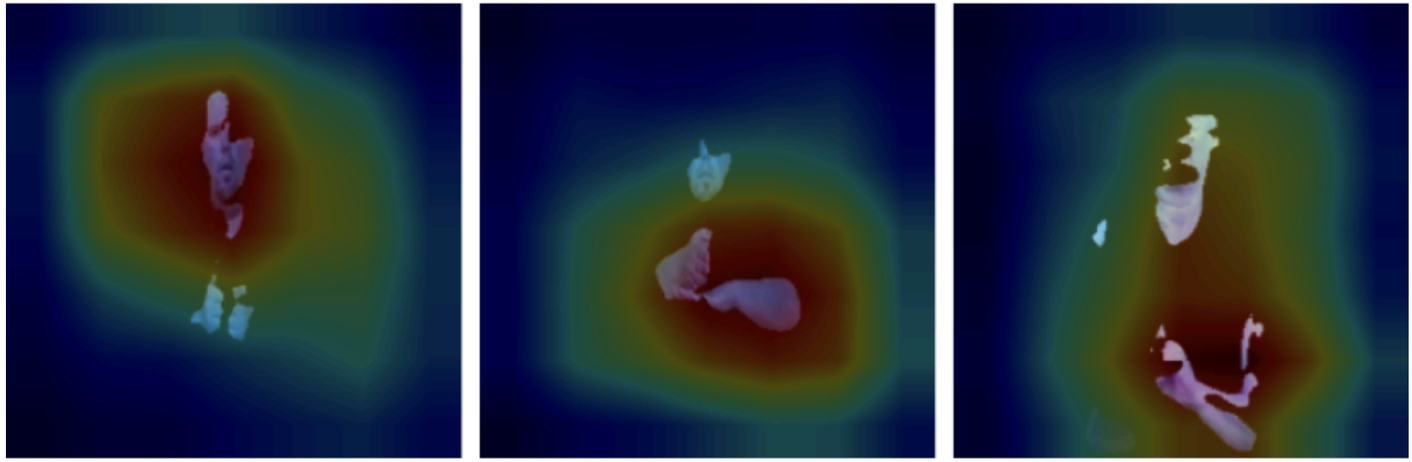
EXPERIMENTAL DESIGN

- **Libraries -** PyTorch
- **Subset -** 50 classes.
- **Dataset split ratio -** (80:10:10)
- **Number of frames -** from the middle 60% of the video.
- **Epochs -** 50
- **Augmentation Technique -** Flip & Rotate
- **Segmentation Models -** U-net, DeepLabV3, Mask R-CNN, E-Net, HRNet, YOLOv8m-Seg
- **Training Setup -** Early Stopping with Patience value 3, dropout, weight decay
- **Hardware used -** Kaggle P100 GPU, Google Colab A100 GPU, and Nvidia-RTX4060 GPU
- **Classification CNN for Label Prediction -** ResNet-18 (Residual Network)
- **Transformer for Label Translation -** Helsinki-NLP (EN -> HI)
- **Evaluation Metrics for CNN Classification -** All Classification Metrics
- **Evaluation for Label Translation -** BLEU score

U-NET

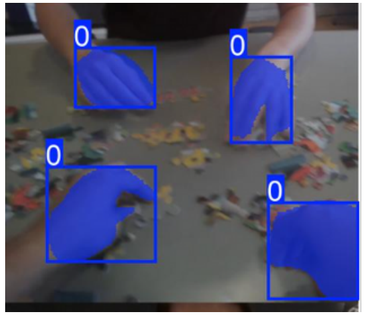
U-Net Specific workflow

- Started with a classic U-Net encoder–decoder (4 encoder blocks - bottleneck - 4 decoder blocks) with skip connections to retain fine spatial details.
- Incorporated BatchNorm + spatial Dropout (0.2 in conv blocks, 0.3 in bottleneck) and Adam + weight-decay (1e-5) for strong regularization.
- Used a final 1×1 convolution to predict two channels (background vs. hand) at 256×256 resolution.
- Trained for 50 epochs on EgoHands with augmentations and monitored pixel-accuracy & AUC-ROC to detect overfitting.
- Added pre/post-training sanity checks (forward-pass shape assertions and save/load consistency) to ensure model integrity and reproducibility.



Accuracy	0.63
Precision	0.81
Recall	0.64
F1	0.69

YOLOv8m-seg

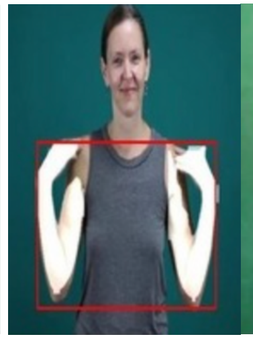


- **YOLO Segmentation Training Results:**

The dataset, originally annotated with hand polygons, was converted into YOLO-style .txt files containing both bounding box coordinates and segmentation information.

Map@50 (IoU)	: 0.9866
Map precision	: 0.9844
Map Recall	: 0.9595

True: drama | Pred: drama



True: hello | Pred: hello

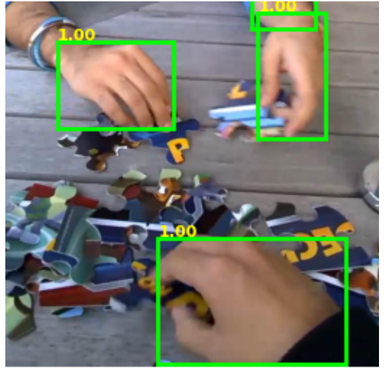
Trained a segmentation model to detect hands using YOLOv8m-Seg. Ran the trained model on WLASL images to extract hand masks and bounding boxes.

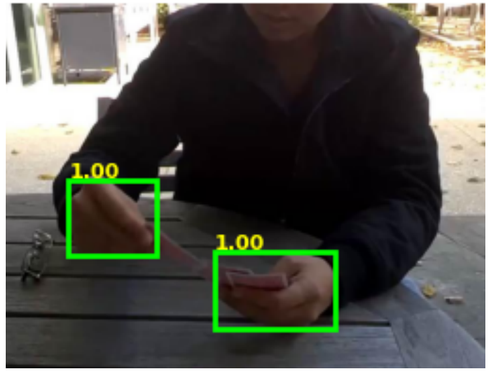
- **ResNet-18 Classification Results:**

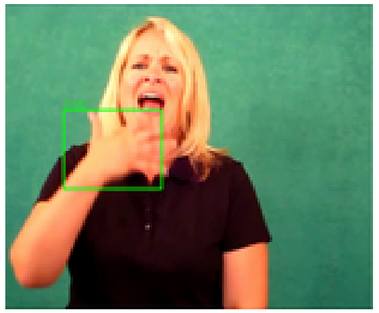
Segmented WLASL images are passed to ResNet-18 for sign classification.

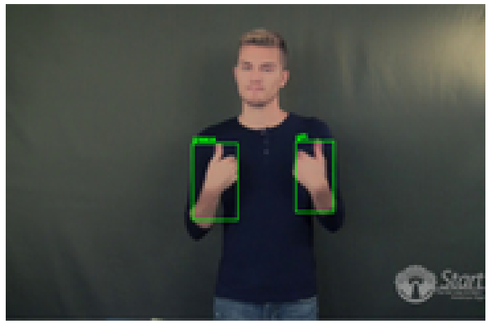
Accuracy	0.8896
Precision	0.9173
Recall	0.8896
F1	0.8844

MASK R-CNN







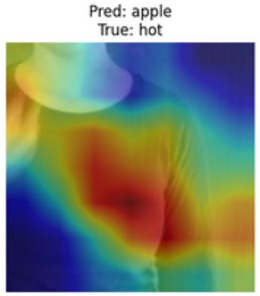


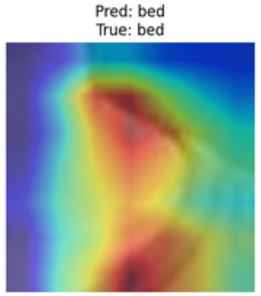
EgoHands:

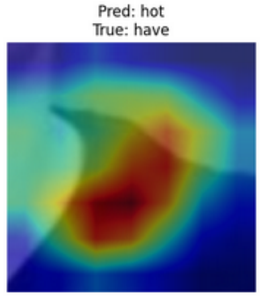
- Trained a Mask R-CNN segmentation model on EgoHands images to accurately detect and segment hands at pixel-level.
- Used originally as it provides polygon annotations for hands.
- Converted these annotations into the COCO JSON format for training Mask R-CNN.

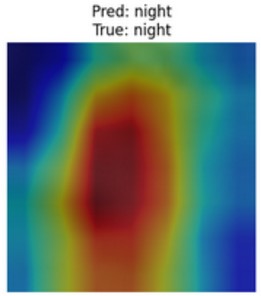
WLASL:

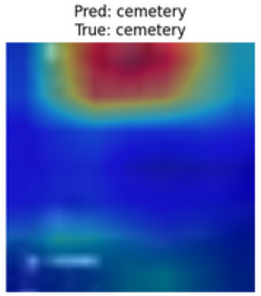
- Fine-tuned the EgoHands-trained Mask R-CNN model on labeled WLASL frames to improve precise hand detection specifically for gesture recognition in sign-language videos.
- Main annotation file containing video URLs, labels (gesture words).











Accuracy	0.534
Precision	0.899
Recall	0.389
F1	0.54

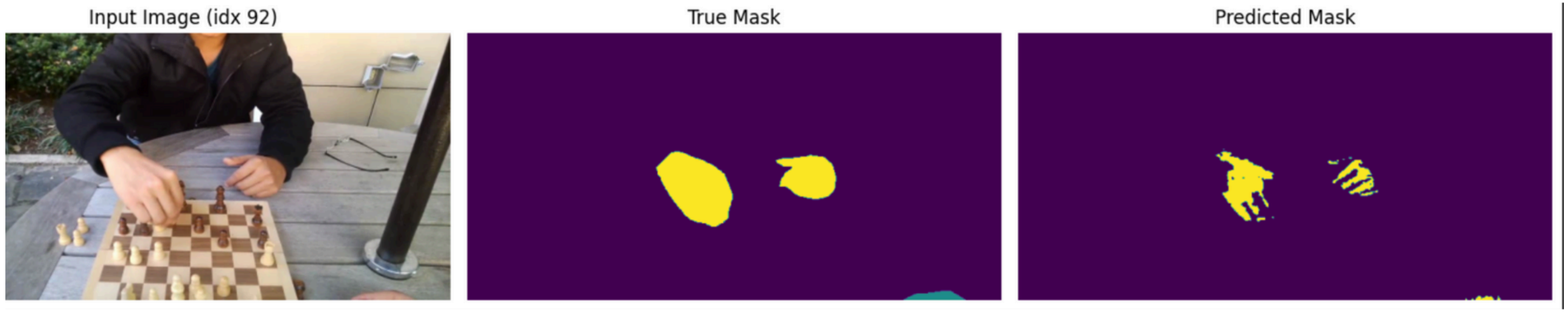
HRNet

HRNet Specific workflow

- Used the powerful pre-trained HRNet-W18 as the starting point.
- Took advantage of HRNet's ability to keep detailed, high-resolution information throughout the network.
- Added a simple Convolutional layer on top to predict the 2 output classes (background, hands).
- Resized the model's prediction map to match the input image size for the final segmentation.

HRNet Results (50 Epochs)

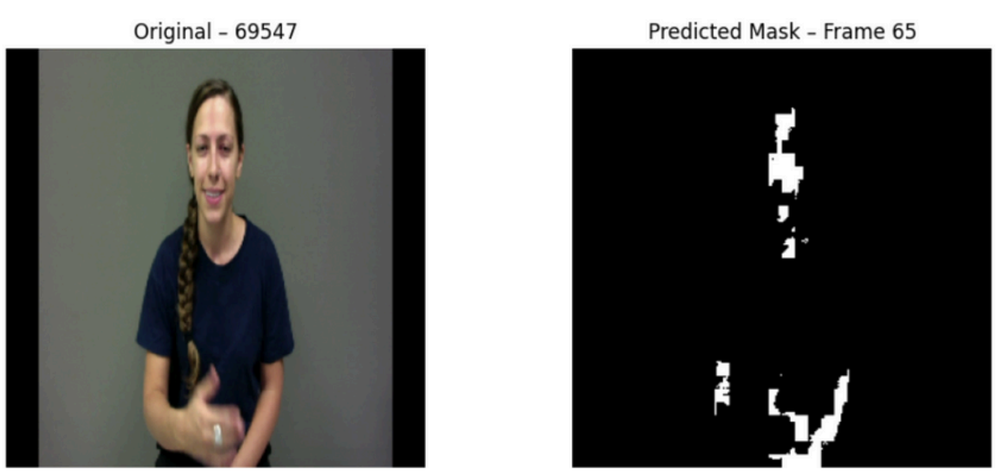
- Training: Model successfully trained on corrected masks.
- Visuals: Model correctly identifies and predicts hand regions (detecting Classes 1 & 2), but segmentation boundaries are coarse.

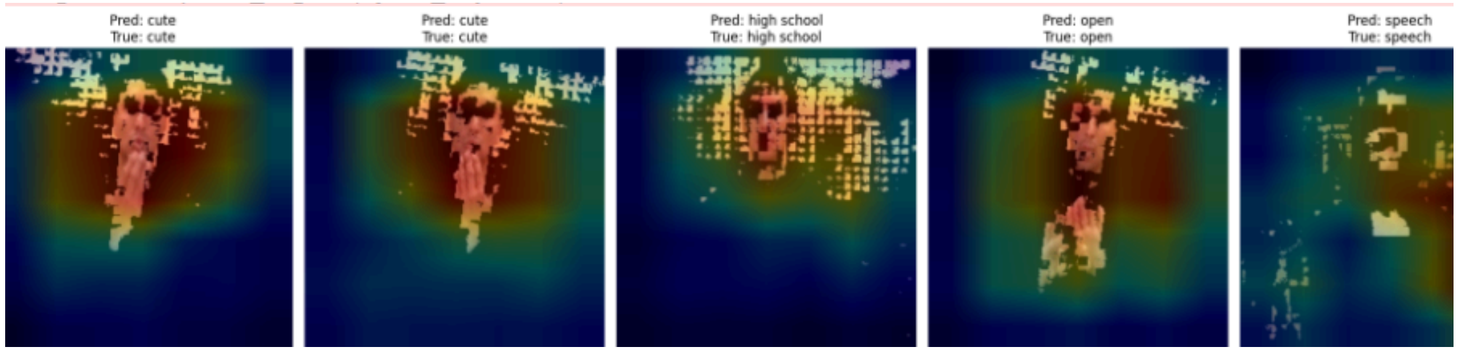


Accuracy	0.86
Recall	0.65
Precision	0.82
F1	0.79

E-NET

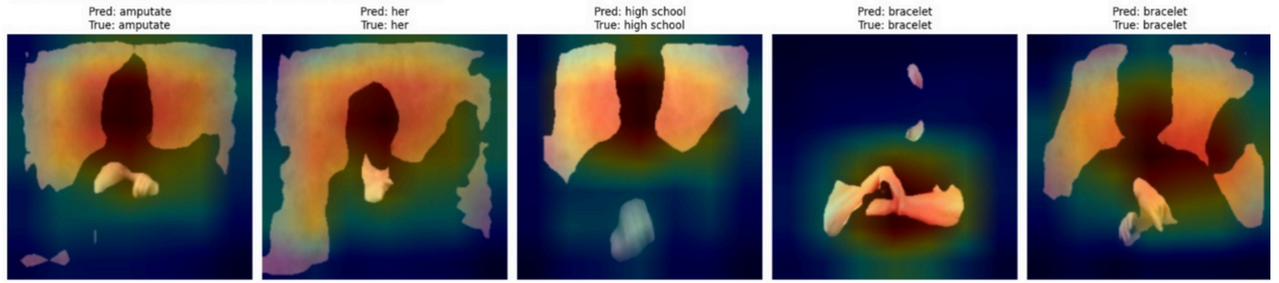
- Leveraged the E-Net encoder-decoder design with early downsampling, asymmetric convolutions, and lightweight bottlenecks for fast, memory-efficient segmentation.
- Fine-tuned the model on 500 EgoHands samples, using BCE+Dice loss to balance class imbalance and contour sharpness.
- Applied random crops, flips, and brightness shifts for augmentation
- Output binary masks were post-processed and used for training a ResNet18 classifier on 50 glosses.
- Achieved pixel-level hand detection suitable for real-time gesture workflows and integrated with Grad-CAM++ visualization for interpretability.





Accuracy	0.94
Recall	0.94
Precision	0.96
F1 Score	0.94

DeepLabV3



- **Preprocessing for EgoHands (Segmentation Training)**

  Converted polygons to binary masks (1 = hand, 0 = background)

  Resized all images and masks to 256×256

  Normalized RGB images with ImageNet mean/std

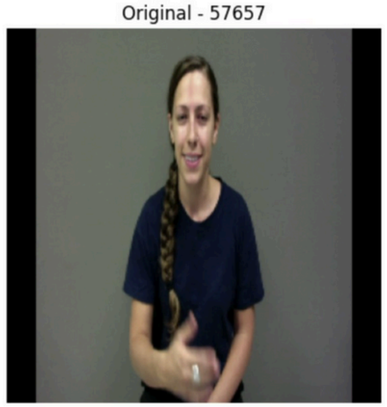
- **Preprocessing for WLASL Videos (Hand Segmentation)**

Normalized like before (ImageNet)

Used your trained DeepLabV3 to predict masks - it just labels it.

Saved only frames with meaningful hand masks (pixel count > 5000)

Applied the mask to RGB frames → hand-only images



Accuracy	~ 0.96
Precision	0.90
Recall	0.91
F1	0.90

RESULTS COMPARISON

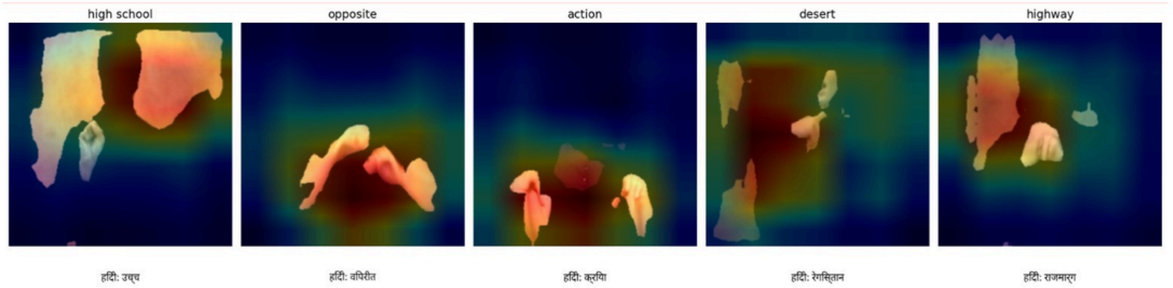
MODEL	U-Net	YOLO	Mask R-CNN	HRNet	E-Net	DeepLab
Accuracy	0.63	0.8896	0.53	0.86	0.94	0.96
Precision	0.81	0.9173	0.89	0.82	0.96	0.90
Recall	0.64	0.8896	0.38	0.65	0.94	0.91
F1	0.69	0.8844	0.54	0.79	0.94	0.90

- **DeepLabV3** achieves the best overall segmentation performance in terms of accuracy and F1-score, making it the top choice for precise hand detection. DeepLab lead in overall effectiveness, achieving the highest accuracy (0.96).
- **E-Net** also performs competitively across all metrics.
- **YOLOv8m-seg** remains a strong alternative for real-time applications due to its speed and competitive performance. YOLOv8m shows a better precision (0.9173).

TRANSLATION

- **Helsinki-NLP/opus-mt-en-hi** a MarianMT-based English-to-Hindi translation model from Hugging Face
- Lightweight (~300MB) and easy to integrate
- Fast inference, even on CPU
- Integrated with gloss classifier output to produce Hindi translations for predicted signs
- smoothing and fuzzy matching (Levenshtein similarity ≥ 0.8) to count semantically similar words as correct.
- Accuracy : 78%

Explored model -
ai4bharat/indictrans2-en-indic-1B
(Huggingface model for IndicTrans2)
Issue: RAM limitations on Colab/Kaggle




```
Gloss: ring
Ref : अंगूठी
Pred: अंगूठी
Exact Match: True
BLEU-1 Score: 1.00
```

```
Gloss: open
Ref : खोलना
Pred: खोलें
Exact Match: False
BLEU-1 Score: 0.00

Gloss: relax
Ref : आराम
Pred: आराम करें
Exact Match: True
BLEU-1 Score: 0.50
```

```
Gloss: remove
Ref : हटाना
Pred: मिटाएँ
Exact Match: False
BLEU-1 Score: 0.00
```

CONCLUSION & FUTURE WORK

Successfully built and validated a modular pipeline for ASL understanding, integrating hand segmentation, sign classification, and translation from English to different language.

FUTURE WORK:

- Address overfitting issue
- Implement more classes
- Real-time implementation
- Implement IndicTrans2 for translation Model
  - State-of-the-art for Indian languages
  - Support for script handling, and multilingual capabilities
  - How? → Use quantization on ai4bharat/indictrans2-en-indic-1B, from 32-bit to 8-bit.

THANK YOU!