Handle: An Open Source Tool For Accurate Gesture Recognition In Real-Time Systems Across Static And Dynamic Modalities

Aadarsh and Raahul

github.com/aadarshjha/handle

Project Goal

- **Goal:** Utilize large scale *hand gesture* databases to refine existing models of *hand gesture recognition* (HGR) to demonstrate state-of-the-art approaches via a deployed application, from static to dynamic.
    - **Experimentation** — blend different approaches to understand why certain architectures are better than others.
    - **Transparency** — most open source studies of HGR are difficult to use and apply. An application would help from an educational and experimental viewpoint.
    - **Summary** — we hope to provide a benchmark for future studies to reflect on various DL approaches in HGR.

Abstract

- **Motivation**:
  - Human-Computer Interaction
  - Augmented, Mixed, Virtual, and General Extended Reality systems
  - Zhou et. al. demonstrates the importance of interaction techniques in manipulating AR/VR content.

- **Problem**:
  - Deep Learning applied to continuous HGR is a new problem.
  - Lack of real-world datasets presents a challenge.
  - Accurate models in deployable, productionized systems is not yet at optimal performance.

Project Dissemination

Aadarsh	Raahul
Static recognition experimentation	Dynamic recognition experimentation
Static Model Deployment Onto Application	Dynamic Model Deployment Onto Application
Full Stack Development Of Application	User and Stress Testing Of Application
Final Presentation, Final Write-Up, Check-In Presentation	Final Presentation, Final Write-Up, Check-In Presentation
	Creating “mini-fied” Dynamic Datasets

Previous Methods | Static and Dynamic

Static

- Processing of a singular image to classify certain gestures
- SVMs, Nearest Neighbor, Graphs, Linear Embeddings.
- Moving towards CNN, data augmentation, etc.

Dynamic

- Process sequence of images (frames) in a video
- 3D CNN, 3D CNN + LSTM
- Self-attention mechanisms, Vision Transformers applied to video

Data | Static Recognition

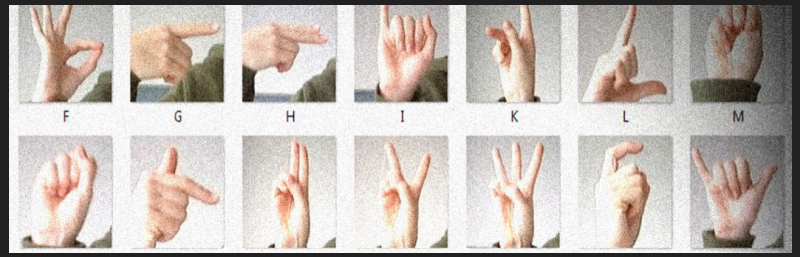
HGRD

-> 10 classes, including: Palm, L, Fist, Fist Moved, Thumb, Index, Ok, Palm, Moved, C, and Down

-> 640 x 240

-> Near Infrared Images

ASLMNIST



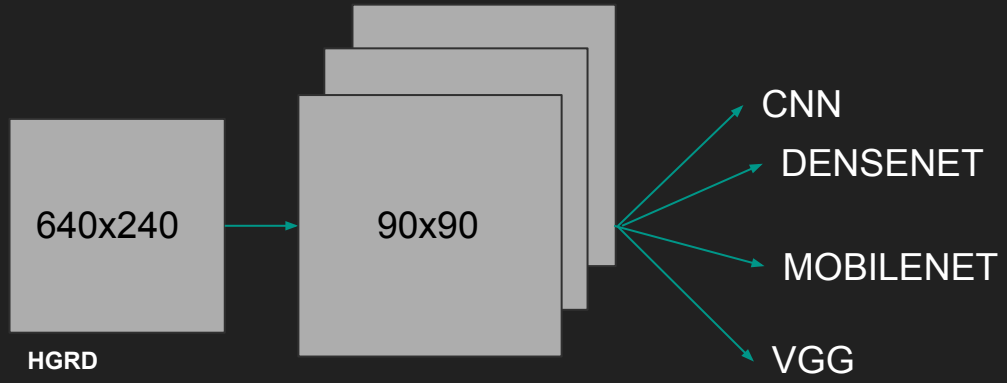
-> 25 classes, alphabetical order

-> 28 x 28

Data | Dynamic Recognition

- MiniIPN
  - Derived from IPN Hand
  - Three classes
    - No gesture
    - Pointing with one finger
    - Double click with one finger
  - Video from front-facing PC web camera
  - 640x480 resolution
  - 30 fps

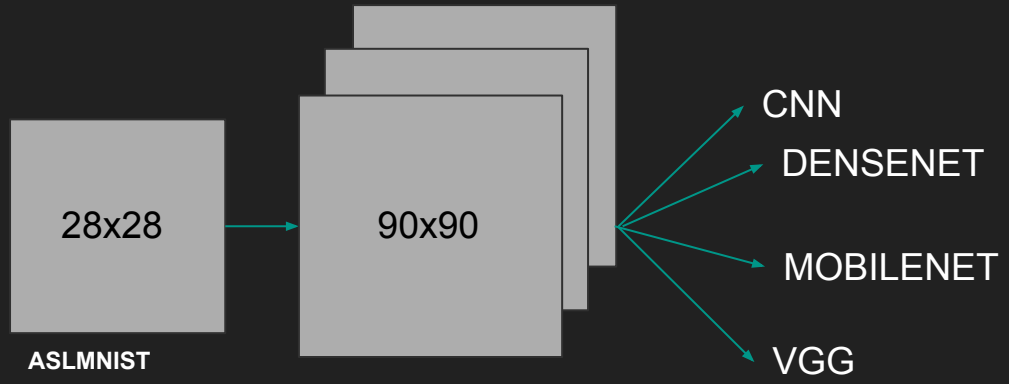
Methods | Static Recognition



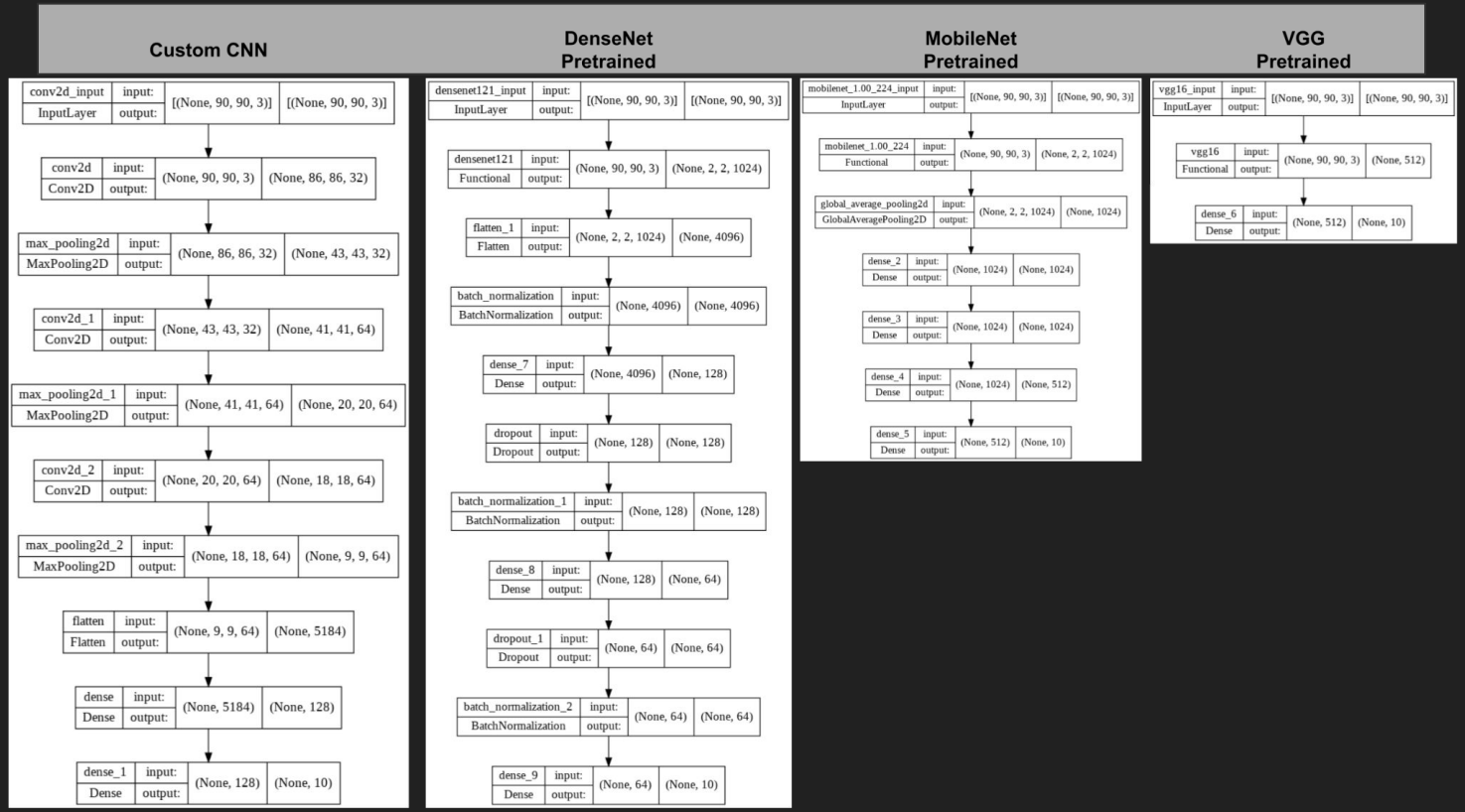
HGRD

```yaml
# set up the hyperparams with YAML
EXPERIMENT_NAME: '5_32_CNN_TEST'
CONFIG:
  NUM_FOLDS: 5
  EPOCHS: 10
  BATCH_SIZE: 32
  VERBOSE: 'false'
  OPTIMIZER: 'adam'
  LOSS: 'sparse_categorical_crossentropy'
  MODE: 'CNN'
```

```yaml
# set up the hyperparams with YAML
EXPERIMENT_NAME: '5_32_CNN_FINAL'
CONFIG:
  NUM_FOLDS: 5
  EPOCHS: 10
  BATCH_SIZE: 32
  VERBOSE: 'false'
  OPTIMIZER: 'adam'
  LOSS: 'categorical_crossentropy'
  MODE: 'CNN'
```

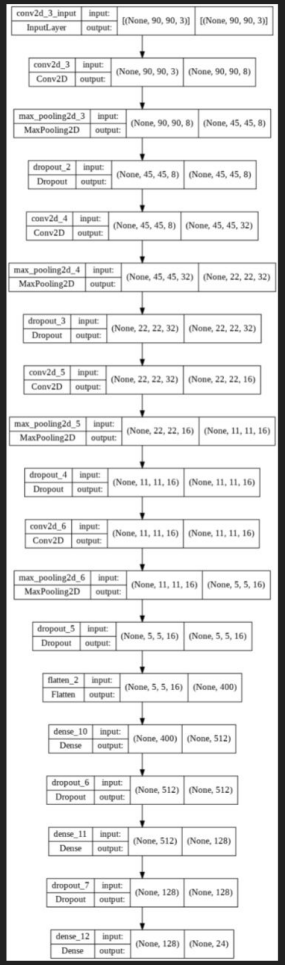


ASLMNIST

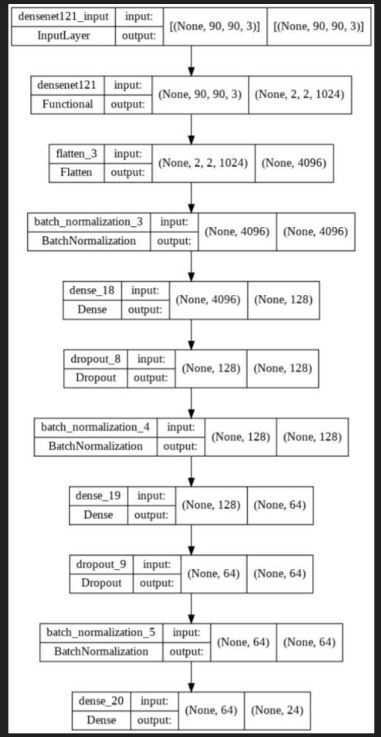


Methods | Static Recognition, Network Figures | HGRD

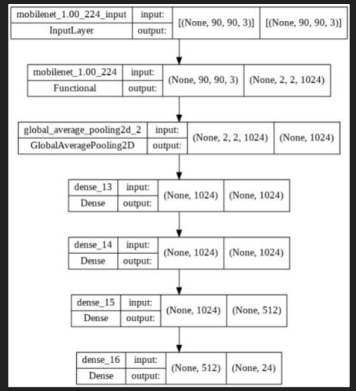
Custom CNN



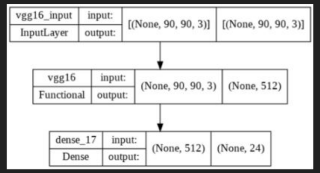
DenseNet Pretrained



MobileNet Pretrained

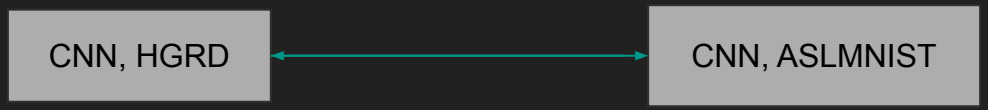


VGG Pretrained



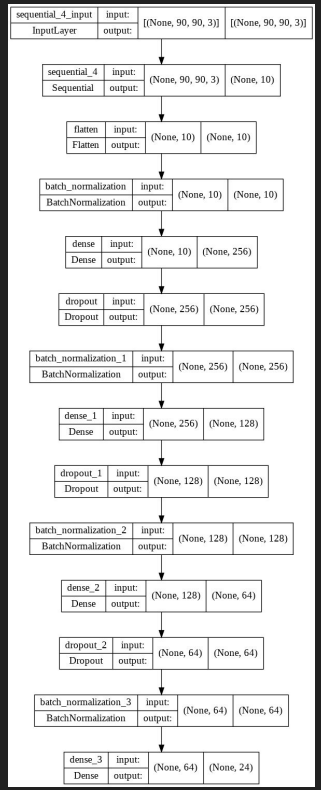
Methods | Static Recognition, Network Figures | ASL

Methods | Static Recognition

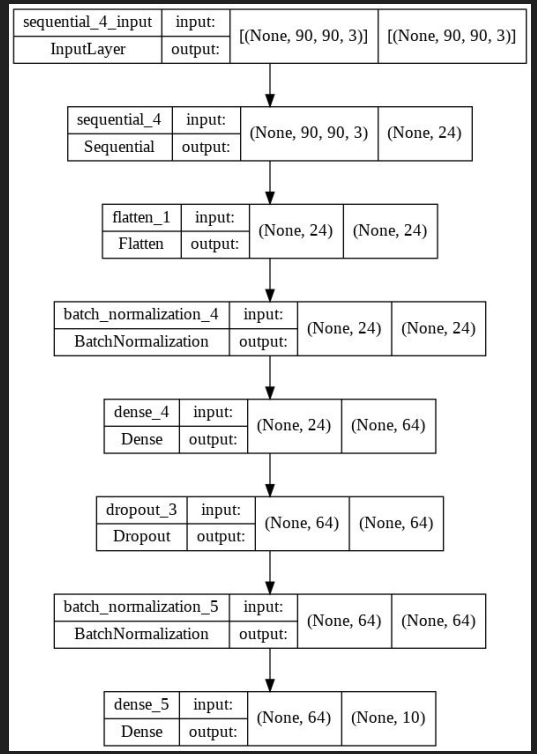


Methods | Static Recognition, Network Figures

HGR transfer



ASL transfer



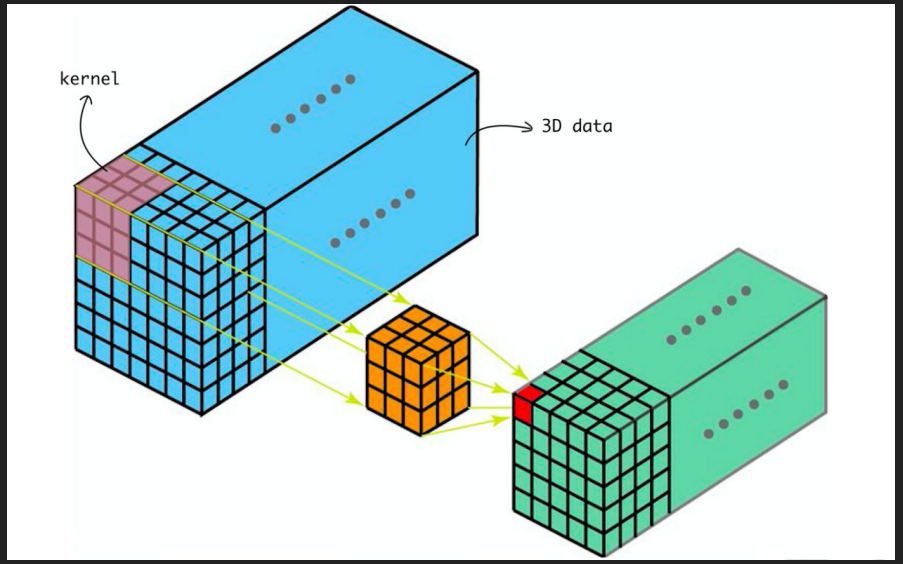
Methods | Dynamic Recognition

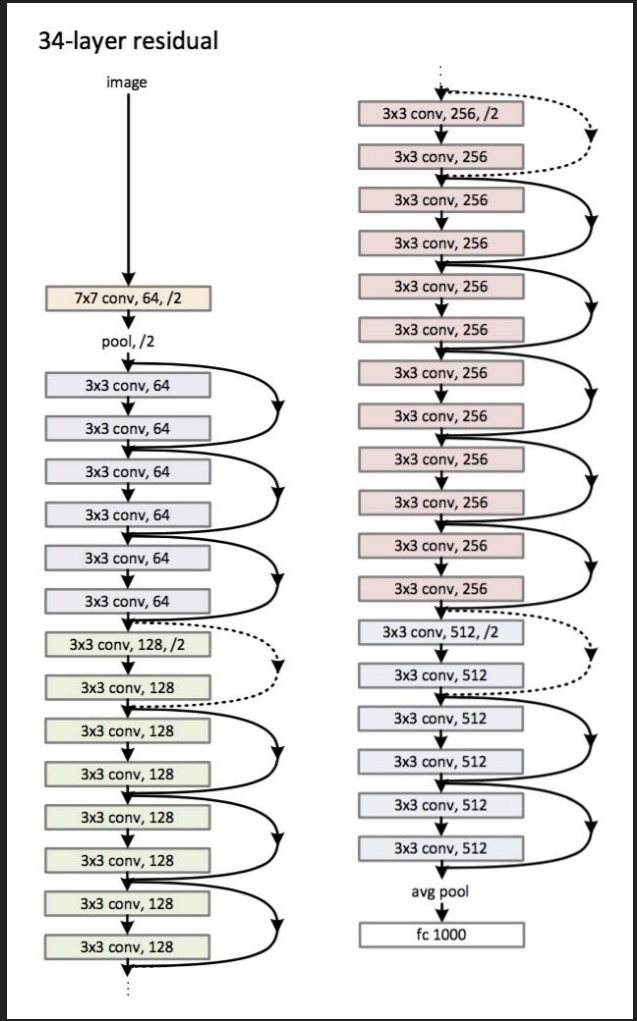
- Create MiniIPN dataset
  - Extract three classes from original IPN Hand Dataset
- Evaluate - training, validation, testing splits
  - IPN pretrained 3D CNN
  - Jester pretrained 3D CNN
  - EgoGesture pretrained 3D CNN
  - Kinetics pretrained 3D CNN
  - ImageNet pretrained CNN-LSTM
  - HowTo400 pretrained TimeSformer

Methods | Dynamic Recognition

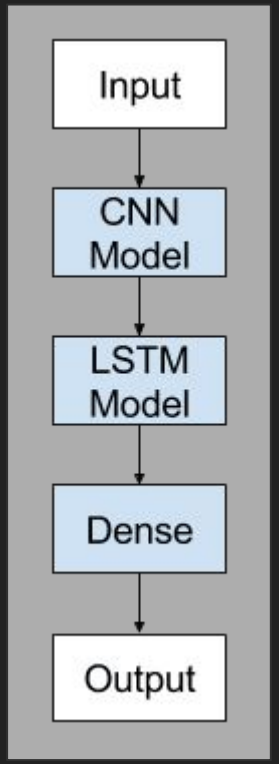
- Configurations
  - Frame size: 112
  - Frame Ct: 32
  - Train for 5 epochs (10 for TimeSformer)
  - Batch size: 8
  - Cross Entropy Loss
  - SGD Optimizer w/ Momentum
  - Learning Rate: 0.001

Methods | Dynamic Recognition



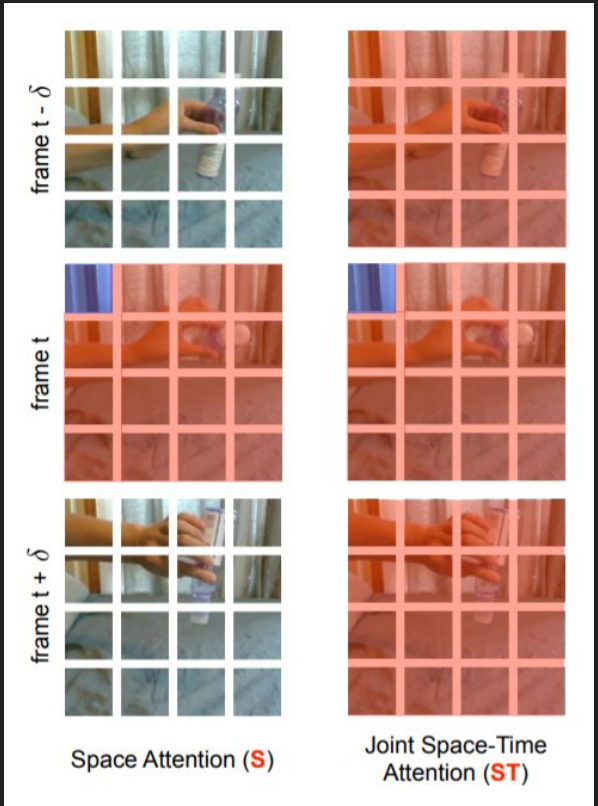


Methods | Dynamic Recognition



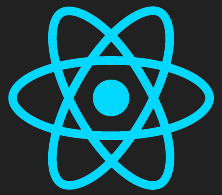
- Idea: CNN-LSTM
  - CNN & LSTM are separate
  - Use CNN as a fixed-feature extractor
    - Can use popular architectures like AlexNet, VGG, ResNet pretrained on ImageNet
    - Use CNNs from static problem
  - Only need to train the LSTM and classification layer

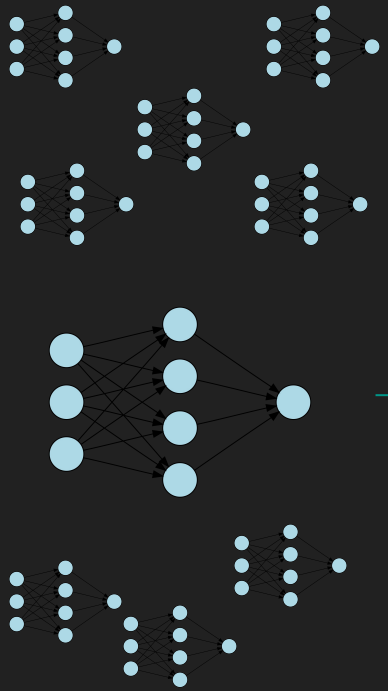
Methods | Dynamic Recognition



- Decompose frame into patches
- Flatten patches into vectors
- Map patch vector into embedding using 2D convolution
- Each patch embedding is passed into Transformer
  - Use blue patch as the query patch in self-attention
  - Red patches are neighbors (other patches to use to calculate attention)
  - Uses more than just the three frames displayed here

Methods | Application





github.com/aadarshjha/handle/tree/master/web

```
Database A, Predictions:
     One – 90%
     Two – 8%
     Three – 2%

Database B, Predictions:
     Pause – 92%
     Fast Forward – 6%
     Rewind – 2%
```

Frontend

Middleware

Backend

Static Recognition | Results

Table 2. The Precision, Recall, F1, and Accuracy scores of HGRD.

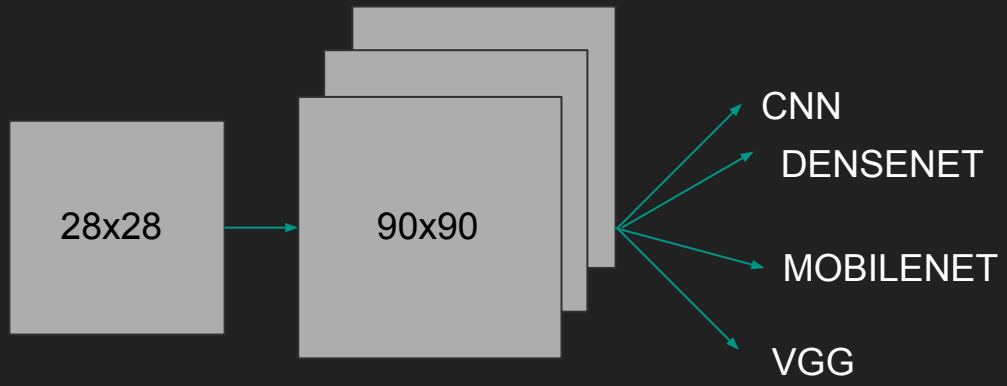
	CNN	DenseNet	MobileNet	VGG
Precision	0.9865	0.9069	0.9692	0.9796
Recall	0.9865	0.8789	0.9670	0.9790
F1	0.9865	0.8845	0.9668	0.9790
Accuracy	0.9865	0.8789	0.9670	0.9790

Table 3. The Precision, Recall, F1, and Accuracy scores of ASLMNIST.

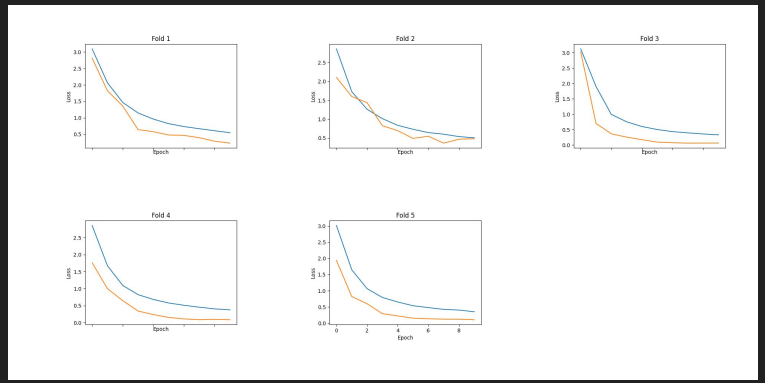
	CNN	DenseNet	MobileNet	VGG
Precision	0.9615	0.9931	0.9837	0.9868
Recall	0.9596	0.9918	0.9820	0.9867
F1	0.9598	0.9917	0.9819	0.9866
Accuracy	0.9596	0.9918	0.9820	0.9867

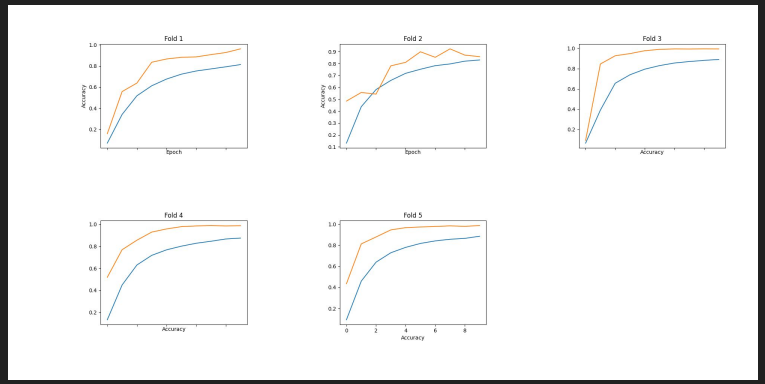
Table 4. The Precision, Recall, F1, and Accuracy scores of domain transfer.

	HGRD Into ASL Model	ASL Into HGRD Model
Precision	0.1857	0.4990
Recall	0.2041	0.3862
F1	0.1409	0.3525
Accuracy	0.2041	0.3862

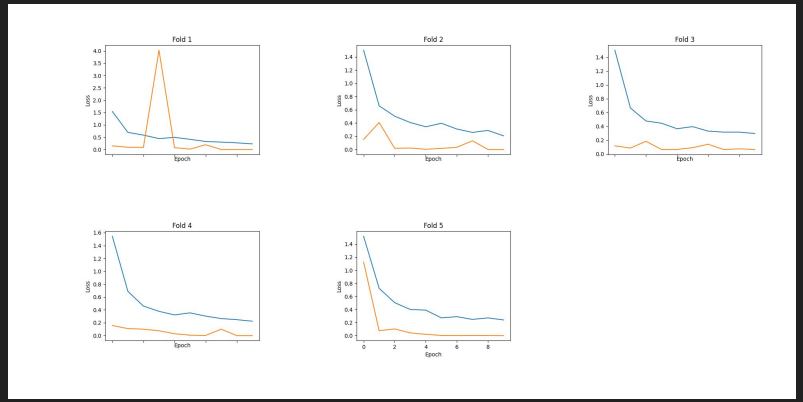

28x28
90x90
CNN
DENSENET
MOBILENET
VGG

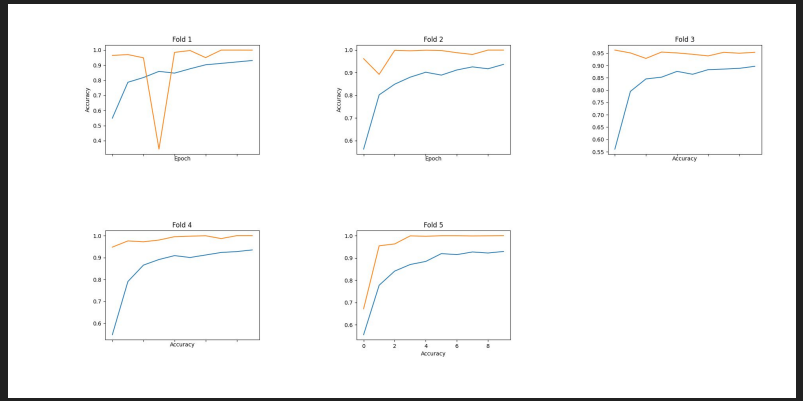
5_32_CNN.yaml, ASL



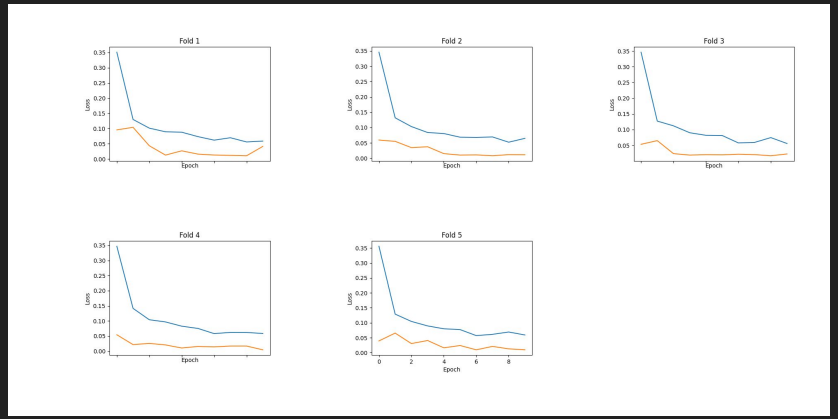


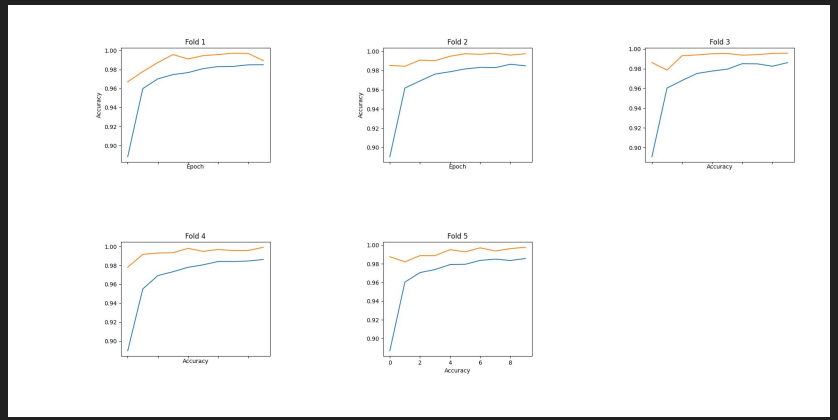
5_32_DENSENET.yaml, ASL



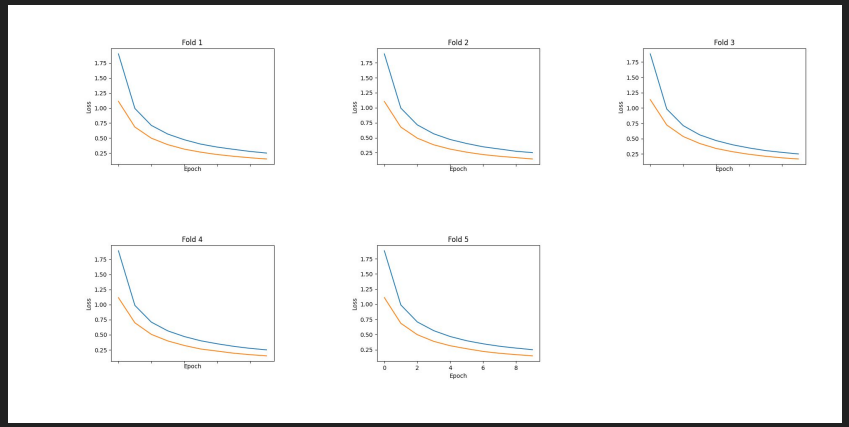


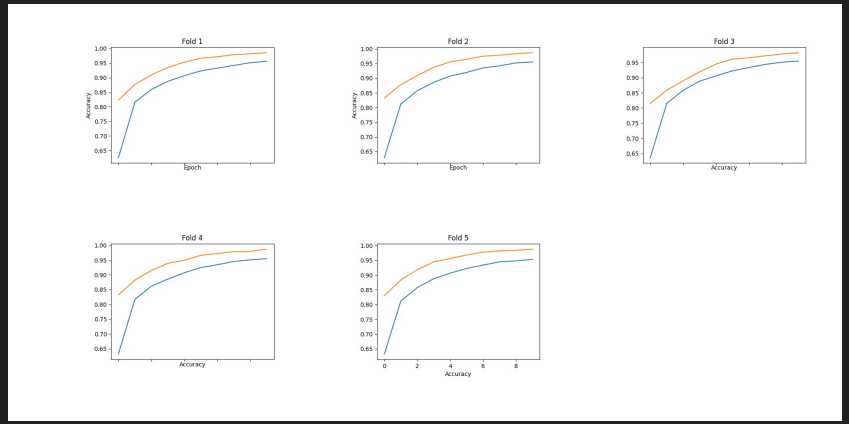
5_32_MOBILENET.yaml, ASL

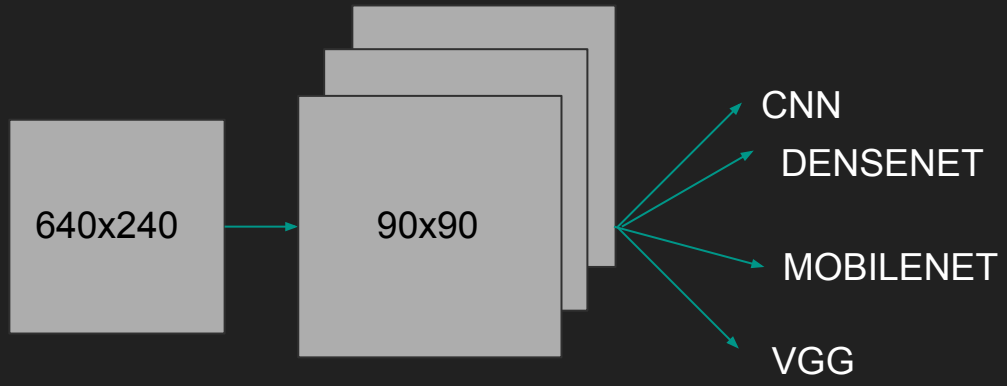




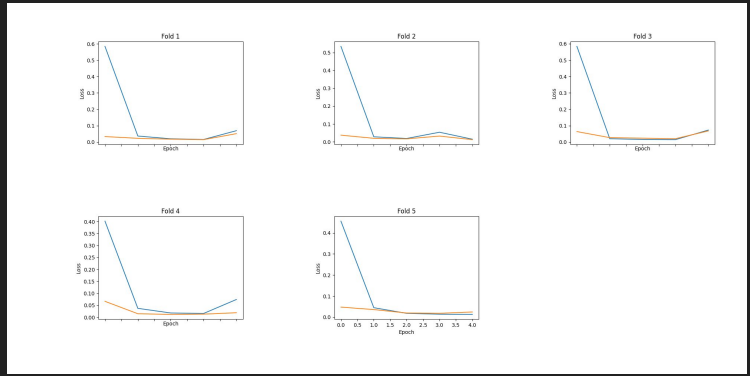
5_32_VGG.yaml, ASL

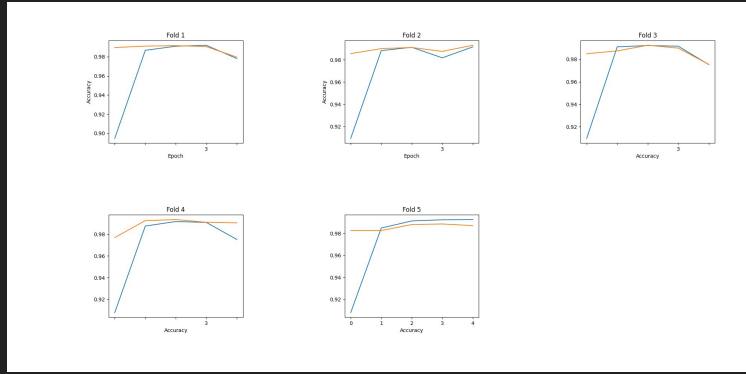




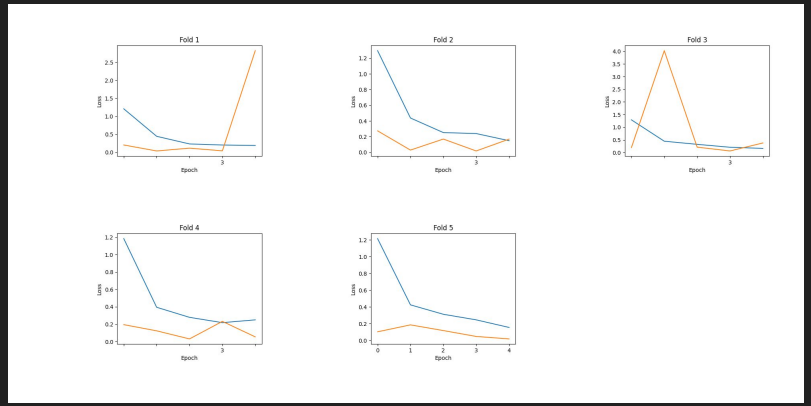

640x240
90x90
CNN
DENSENET
MOBILENET
VGG

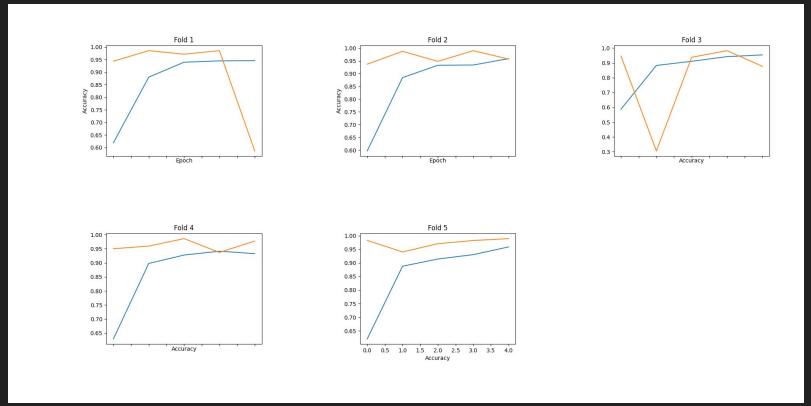
5_32_CNN.yaml, HGRD



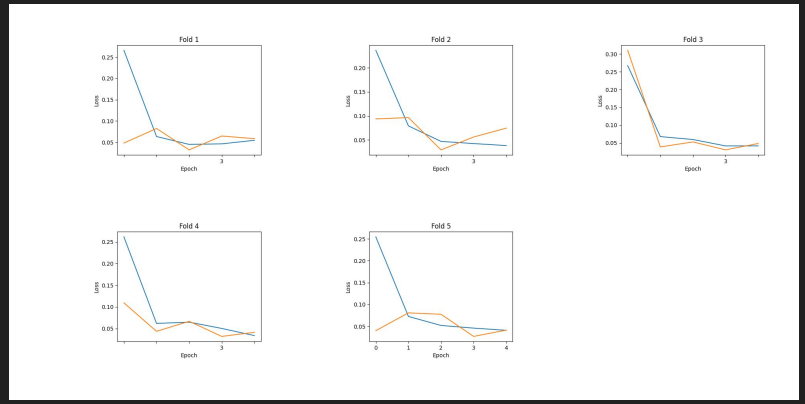


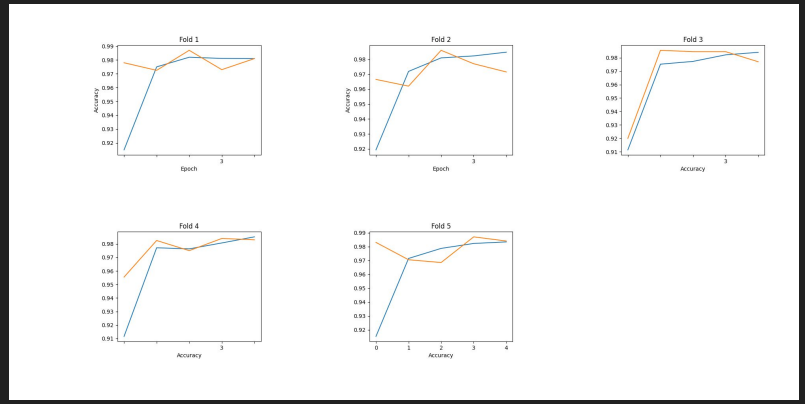
5_32_DENSENET.yaml, HGRD



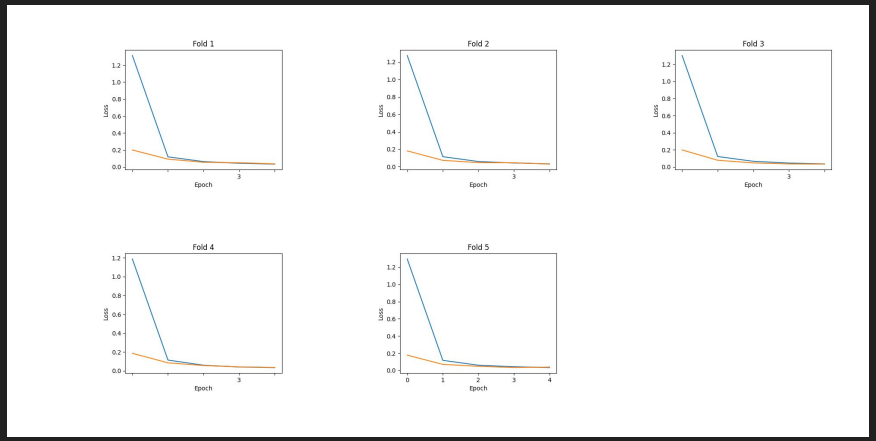


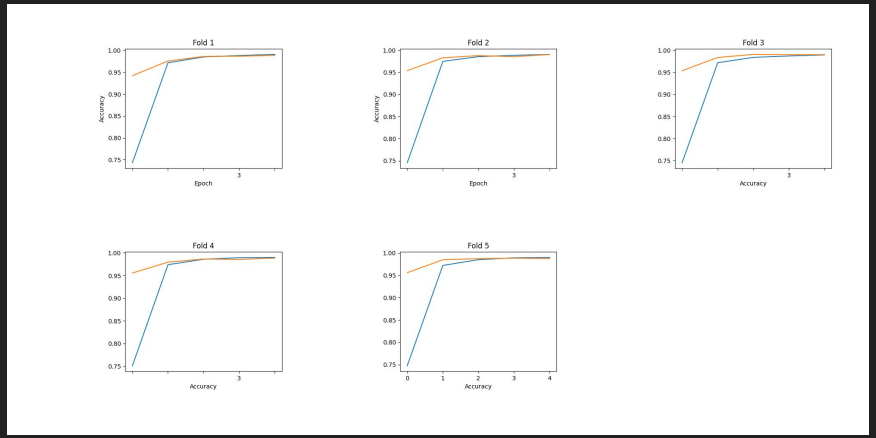
5_32_MOBILENET.yaml, HGRD



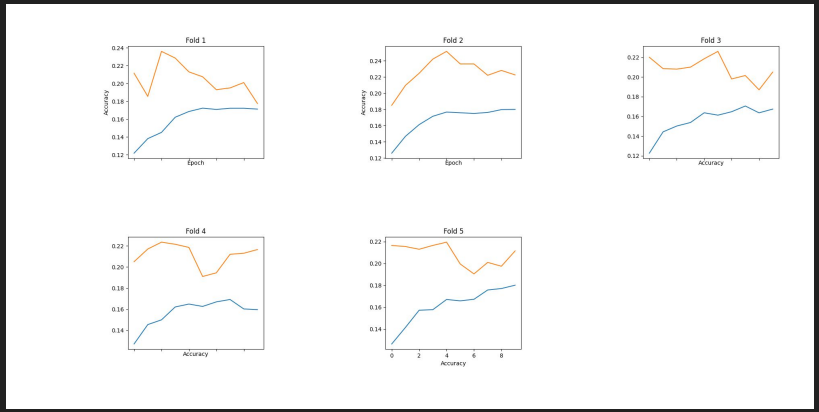


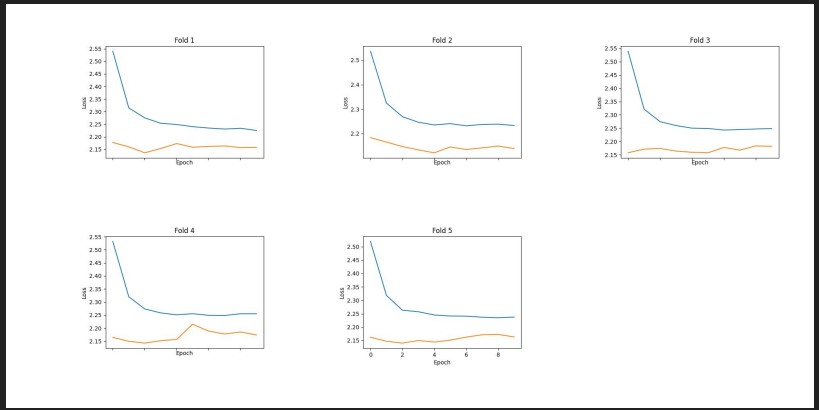
5_32_VGG.yaml, HGRD



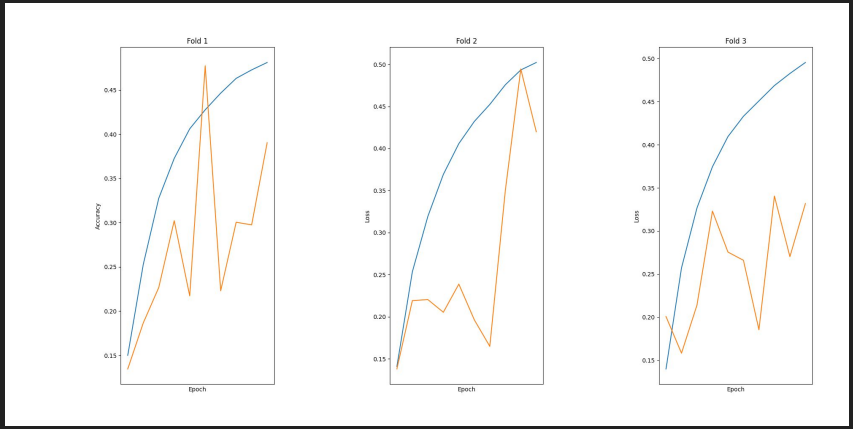


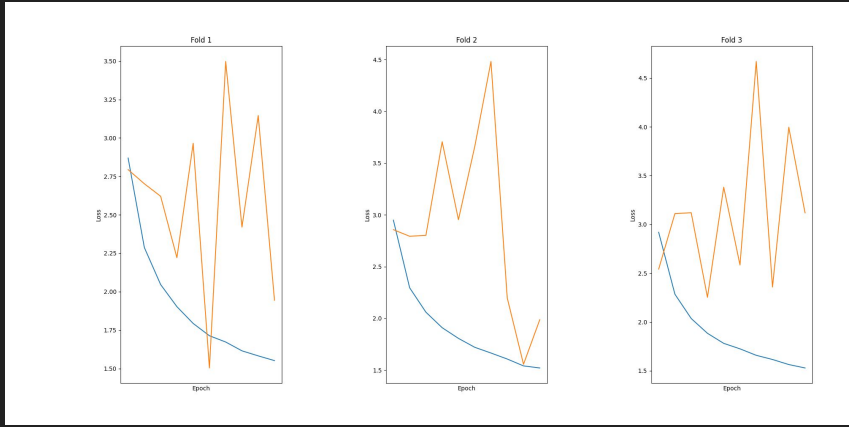
Passing the HGR Data into ASL Model





Passing the ASL Data into HGR Model



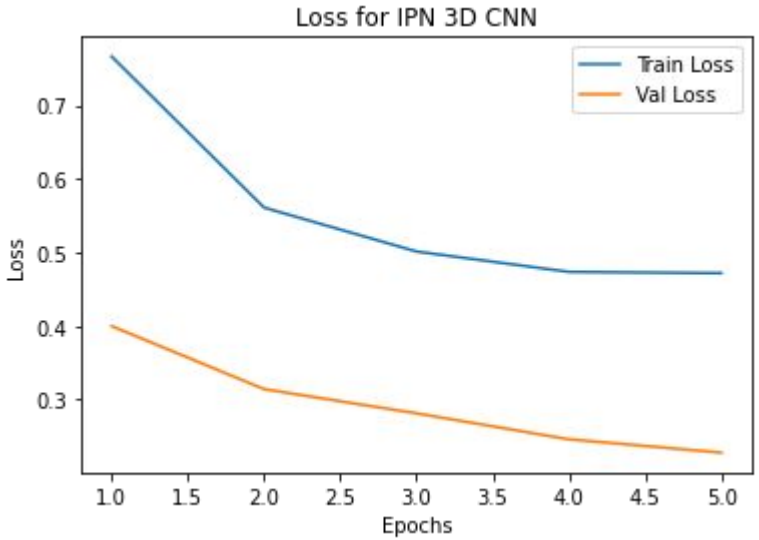


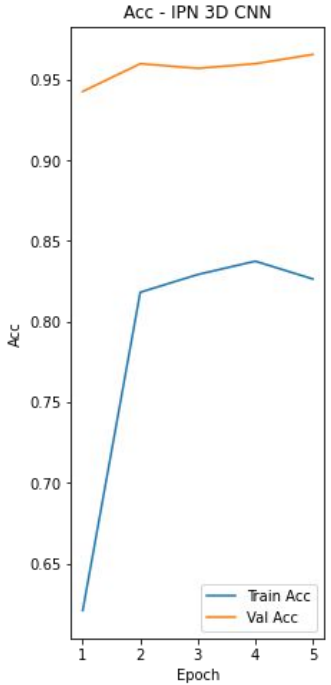
Dynamic Recognition | Results

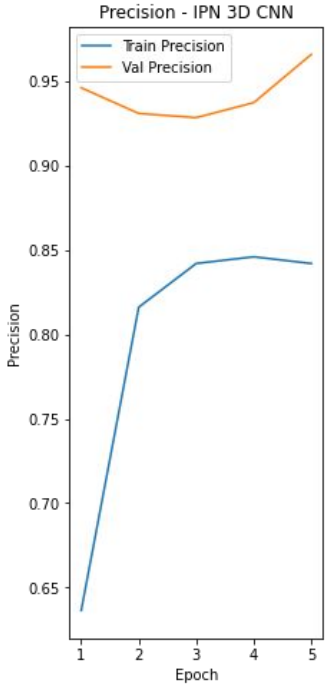
Table 5. The Accuracy, Precision, Recall, and Balanced Accuracy (BACC) scores on MiniIPN.

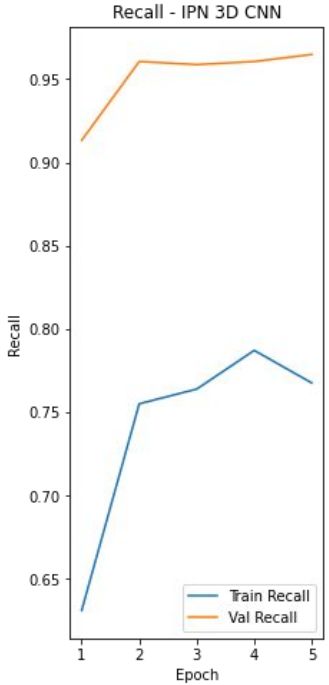
Model	Accuracy	Precision	Recall	BACC
IPN 3D CNN	0.91	0.90	0.84	0.84
Jester 3D CNN	0.83	0.55	0.60	0.60
EgoGesture 3D CNN	0.77	0.81	0.67	0.67
Kinetics 3D CNN	0.75	0.50	0.54	0.54
ImageNet CNN-LSTM	0.57	0.19	0.33	0.33
TimeSformer	0.58	0.39	0.42	0.42

Dynamic Recognition | Results, Training Curves

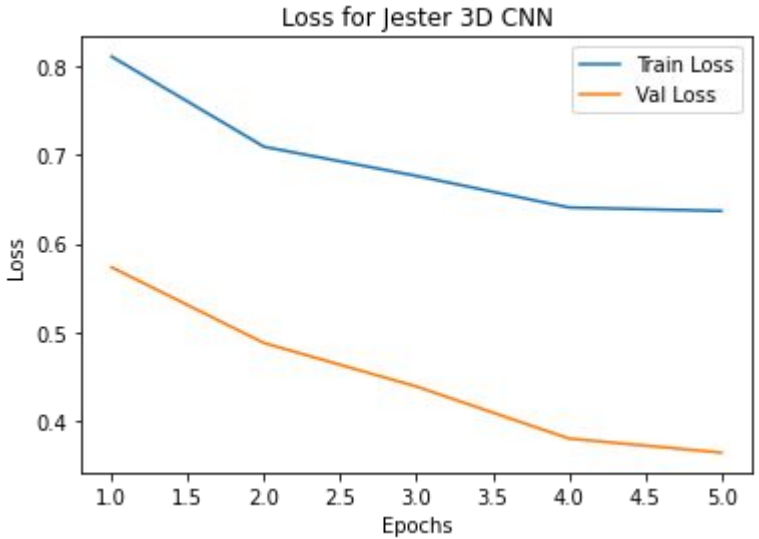


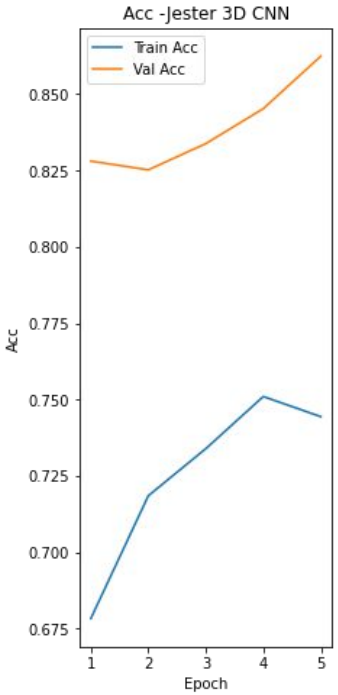


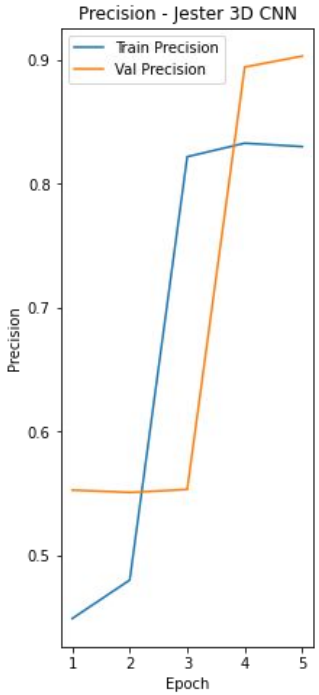


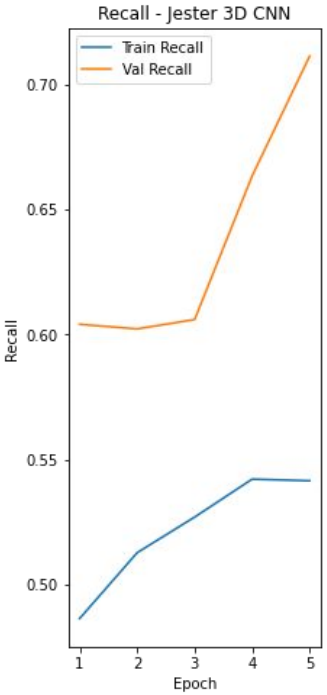


Dynamic Recognition | Results, Training Curves

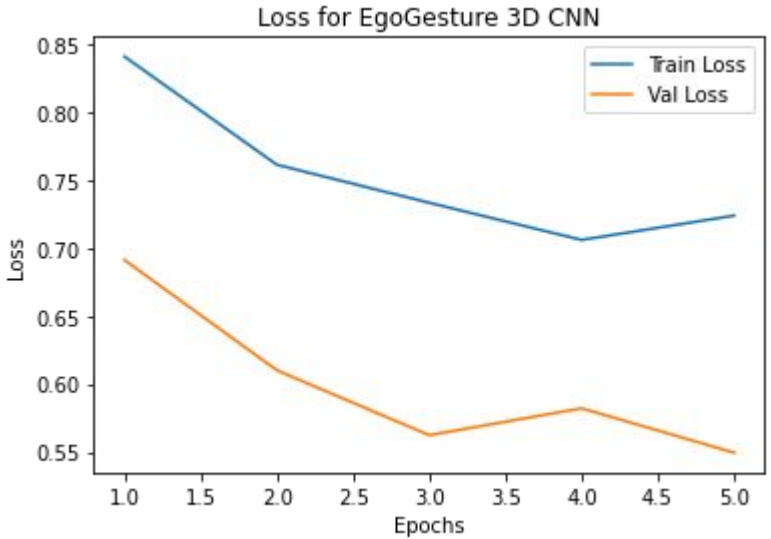


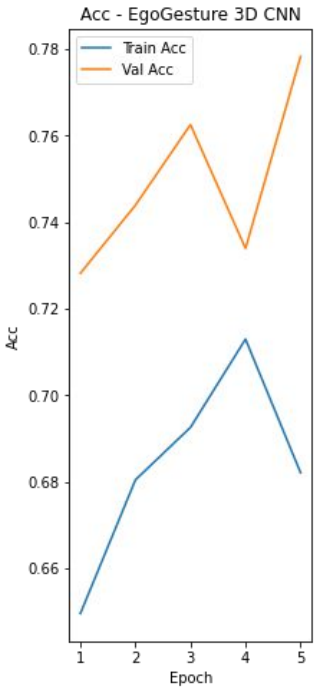


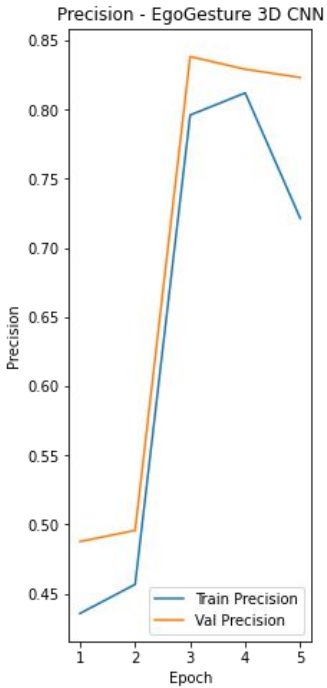


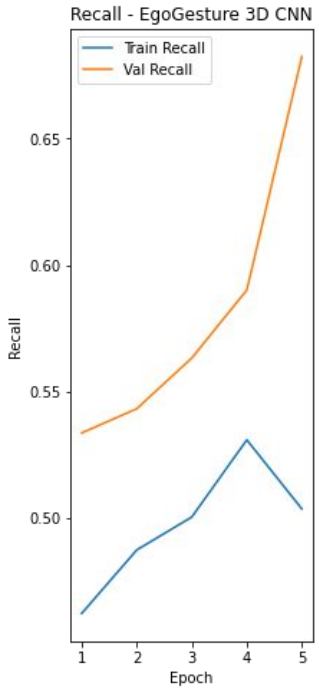


Dynamic Recognition | Results, Training Curves

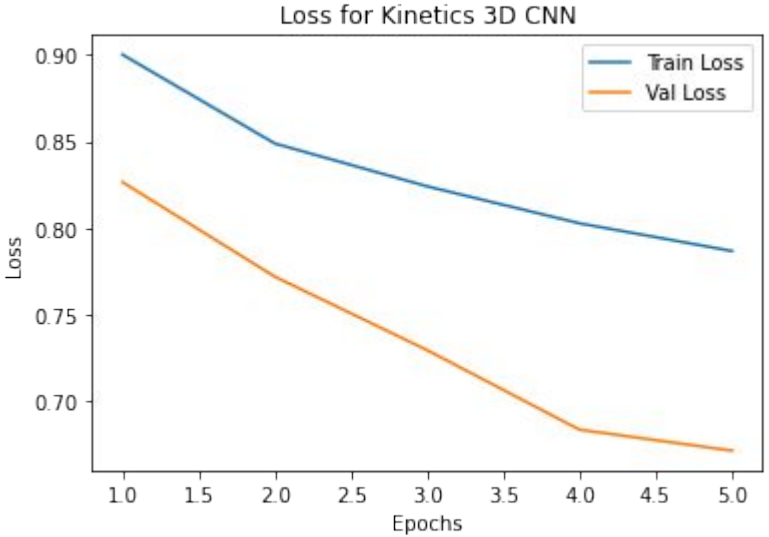


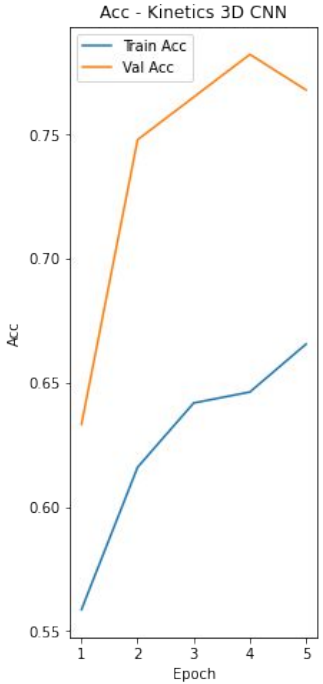


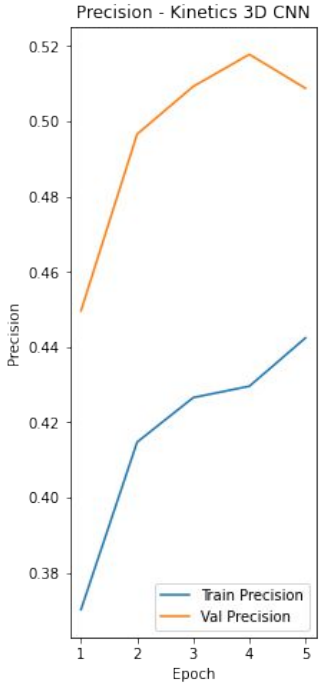


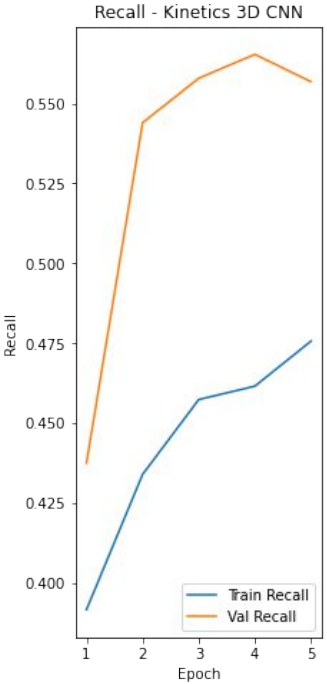


Dynamic Recognition | Results, Training Curves

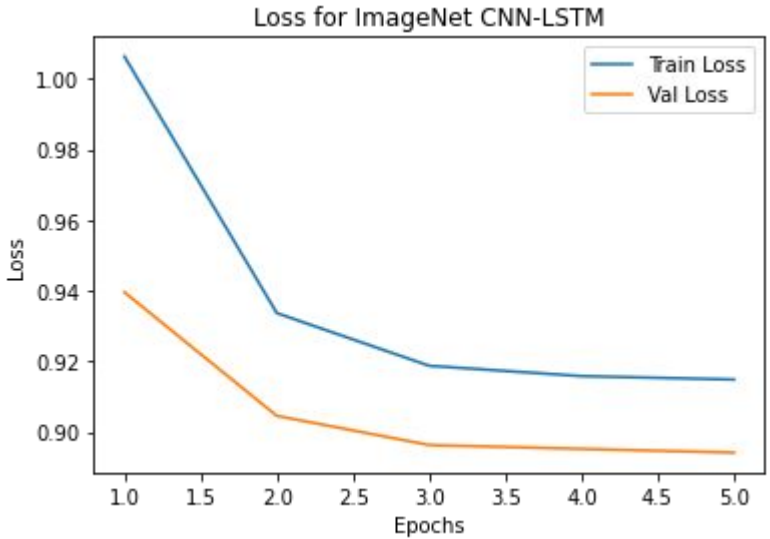


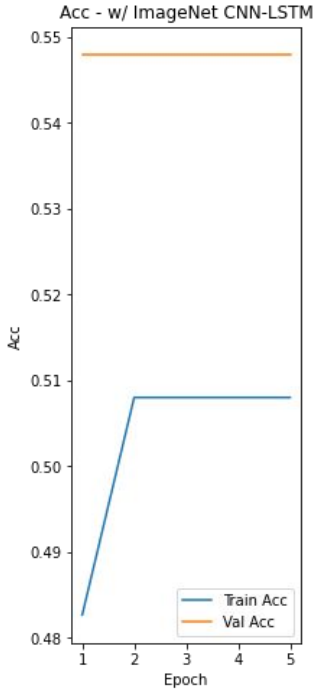


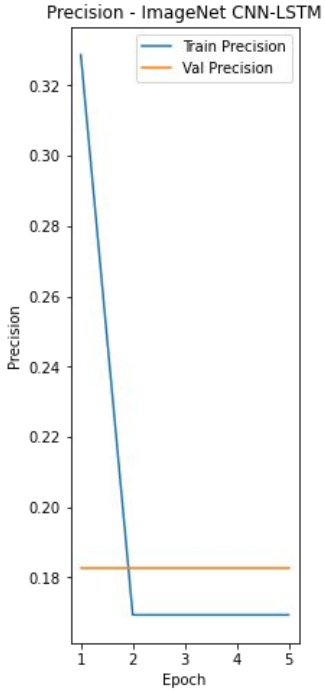


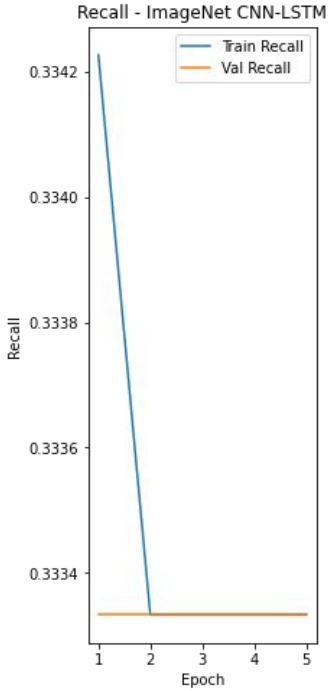


Dynamic Recognition | Results, Training Curves

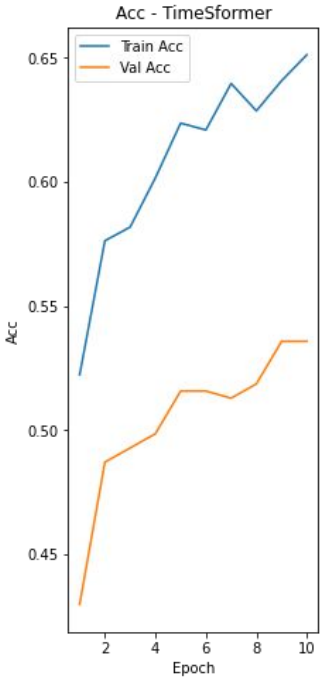


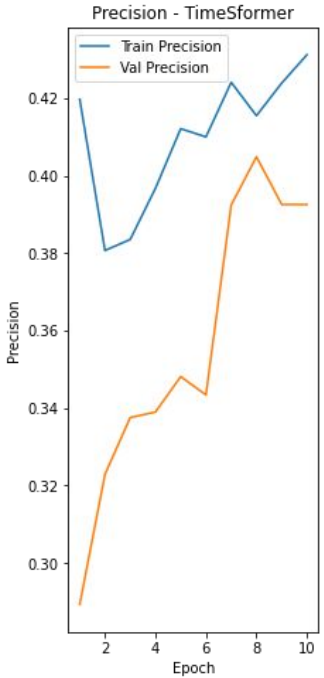


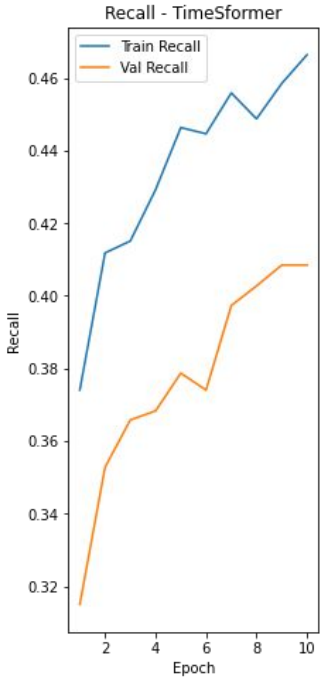




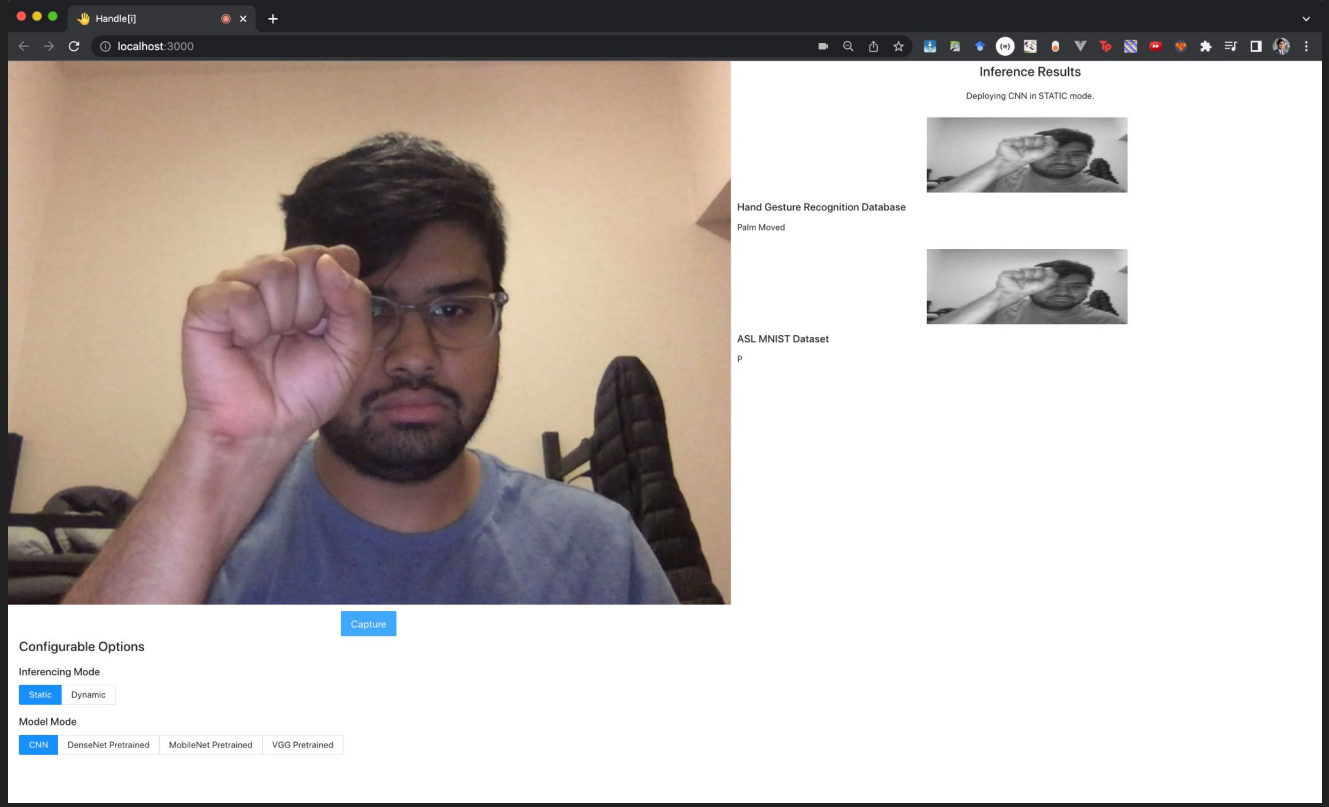
Dynamic Recognition | Results, Training Curves



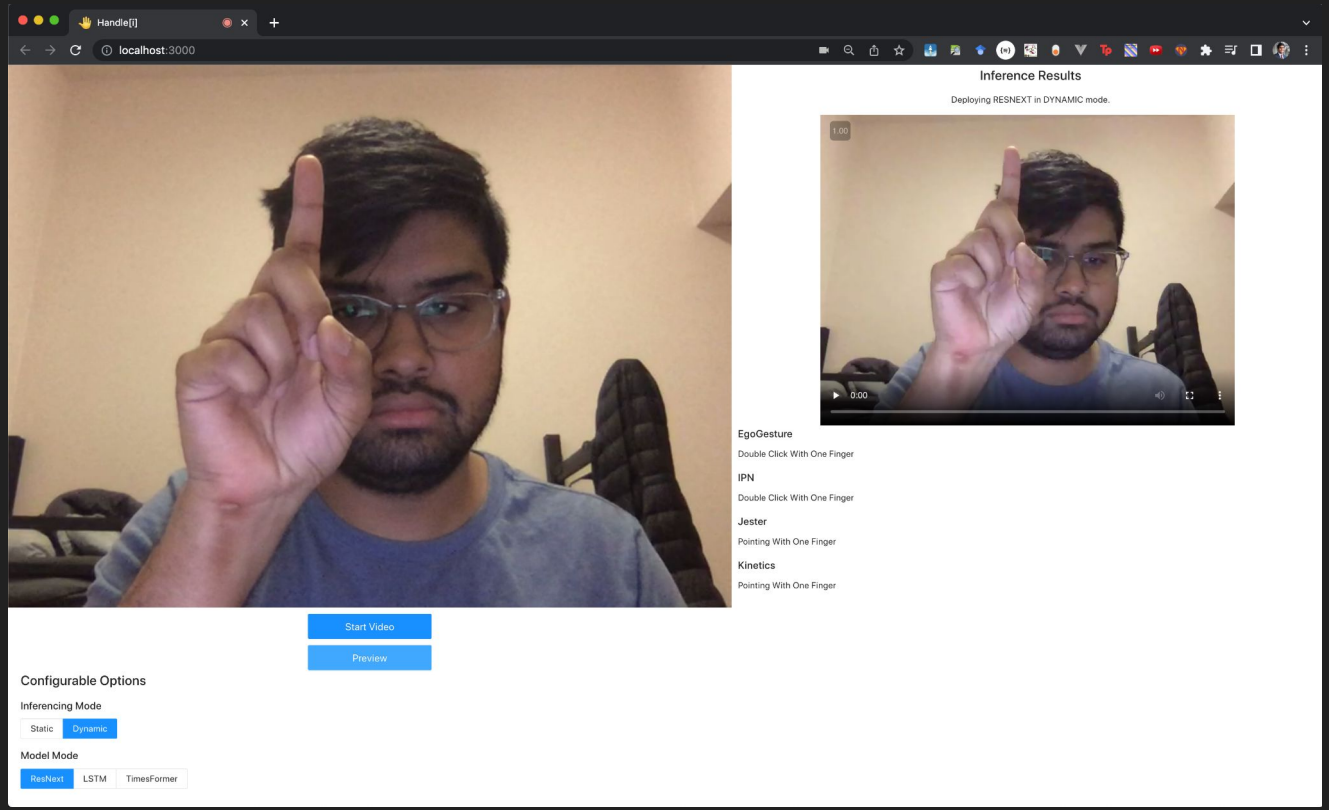




Application | Results / Static



Application | Results / Dynamic



Demo

http://localhost:3000/

Limitations

- Static inference results are weak, in both a transfer learning setting and being deployed to an application.
- Lack of diverse, noisy data.
- Could not fully train a model from scratch on hand gesture datasets for dynamic (computational complexity and data is too high).
- Additional user studies on the application (more formal statistics).

Conclusions

- Static:
  - Consistently strong results are achieved across-the-board (>0.8 for HGRD, and >0.9 for ASL).
  - For HGRD, optimal performance is seen with the custom CNN, with a score of 0.9865.
  - For ASLMNIST, optimal performance is seen with DenseNet, with a score of 0.9918.
  - Poor domain transfer results, 0.2041 and 0.3862
- Dynamic:
  - Strong results on IPN 3D CNN. EgoGesture, Jester, Kinetics 3D CNNs were not the best. CNN-LSTM and TimeSformer performed the worst.
  - IPN 3D CNN - highest accuracy: 94%
- Application
  - Dynamic better than static, overall.

Milestones

- Jan 24th — Project Proposal and Refinement
- Jan 31st — Finalized Topic Idea, Established Roles, Begin Dataset Search
- Feb 7th — Collect and Process Data, Find 2-4 Salable Existing DL Frameworks
  - Reproduce DL Frameworks on Paper Sources
  - Determine If Scope Is Dynamic, Static Or Both
- Feb 14th — Working Understanding Of Frameworks, Propose and Implement Changes
- Feb 21st — Train and Analyze Performance Of Custom Frameworks
- Feb 28th — Buffer Time For Revising Changes
- March 7th — Buffer Time For Revising Changes
- March 14th — Finalize Model and Deploy Onto Secondary and Tertiary Datasets
- March 21st — Collect and Finalize Data, Model, and Framework
- March 28th — Begin Building Real-Time Application
- April 4th — Test Application With Custom Video Of Gestures
- April 11th — Debug Performance Or Latency Issues
- April 18th — Finalize Project, Write Paper and Presentation
- April 25th — Practice and Finalizing Touches

Concurrent

Handle[i]

localhost:3000

Capture

Configurable Options

Inferencing Mode

Static | Dynamic

Model Mode

CNN | DenseNet Pretrained | MobileNet Pretrained | VGG Pretrained

Inference Results

Deploying CNN in STATIC mode.

Hand Gesture Recognition Database

Palm Moved

ASL MNIST Dataset

P

Handle[i]

localhost:3000

Start Video

Preview

Configurable Options

Inferencing Mode

Static | Dynamic

Model Mode

ResNext | LSTM | TimesFormer

Inference Results

Deploying RESNEXT in DYNAMIC mode.

EgoGesture

Double Click With One Finger

IPN

Double Click With One Finger

Jester

Pointing With One Finger

Kinetics

Pointing With One Finger