American Sign Language Detection Report

Objective

Build a system that can detect a given ASL (American Sign Language) input image and output what the sign represents (i.e., which letter of the alphabet or special sign is shown).

Dataset

- **Source**: ASL Alphabet Dataset
- **Download Link**: ASL Dataset
- **Classes**: 29 total
  - 26 for the letters A-Z
  - 3 for SPACE, DELETE, and NOTHING
- **Structure**:
  - **Train set**: Separate folders for each class (alphabet letter or special sign). Each folder contains images for that class.
  - **Test set**: Separate images for each class, named with the class as a prefix.

Data Preparation

- Used `torchvision.datasets.ImageFolder` for the training set, mapping each folder to a class.
- Used a custom dataset for the test set, extracting the class from the filename.
- Split the training set: 85% for training, 15% for validation.
- Applied standard data augmentation and normalization transforms.

Model

- **Architecture**: ResNet18 (transfer learning)
- **Modifications**: Final fully connected layer replaced to output 29 classes.
- **Training**:
  - First, base layers frozen, only final layer trained for 10 epochs.
  - Then, all layers unfrozen and fine-tuned for 5 more epochs.
- **Loss Function**: CrossEntropyLoss
- **Optimizer**: Adam

Training and Validation Results

```
Epoch 1/10 - Train: loss=0.6893, acc=0.8449 | Val: loss=0.2769, acc=0.9328
Epoch 2/10 - Train: loss=0.2529, acc=0.9324 | Val: loss=0.1818, acc=0.9519
Epoch 3/10 - Train: loss=0.1886, acc=0.9470 | Val: loss=0.1329, acc=0.9628
Epoch 4/10 - Train: loss=0.1586, acc=0.9539 | Val: loss=0.1288, acc=0.9618
Epoch 5/10 - Train: loss=0.1420, acc=0.9569 | Val: loss=0.1068, acc=0.9687
Epoch 6/10 - Train: loss=0.1285, acc=0.9604 | Val: loss=0.0946, acc=0.9716
Epoch 7/10 - Train: loss=0.1195, acc=0.9628 | Val: loss=0.0874, acc=0.9724
Epoch 8/10 - Train: loss=0.1153, acc=0.9629 | Val: loss=0.0885, acc=0.9719
Epoch 9/10 - Train: loss=0.1067, acc=0.9659 | Val: loss=0.0948, acc=0.9691
Epoch 10/10 - Train: loss=0.1023, acc=0.9679 | Val: loss=0.0854, acc=0.9732
Fine-tune Epoch 11/15 - Train: loss=0.0178, acc=0.9947 | Val: loss=0.0016, acc=0.9995
Fine-tune Epoch 12/15 - Train: loss=0.0018, acc=0.9996 | Val: loss=0.0007, acc=0.9998
Fine-tune Epoch 13/15 - Train: loss=0.0015, acc=0.9997 | Val: loss=0.0005, acc=0.9998
Fine-tune Epoch 14/15 - Train: loss=0.0011, acc=0.9997 | Val: loss=0.0002, acc=1.0000
Fine-tune Epoch 15/15 - Train: loss=0.0010, acc=0.9997 | Val: loss=0.0001, acc=1.0000
```

Test Results

```
Test: loss=0.0001, acc=1.0000
```

Observations

- The model achieved near-perfect accuracy on both validation and test sets, indicating strong generalization for this dataset.
- Data augmentation and transfer learning were effective in quickly reaching high accuracy.
- The dataset is well-structured and balanced, contributing to the high performance.

Files

- `asl_detection_pytorch.py`: Main training, validation, and testing script.
- `asl_classifier_resnet18.pth`: Final trained model weights.

Future Work

- Test on real-world, non-dataset images to evaluate robustness.
- Explore lighter models for deployment on edge devices.
- Add support for video-based ASL recognition.