Gesture Recognition

Utilizing Deep Learning with Faster-RCNN

By: Tomomi Bahun & Joshua Turner

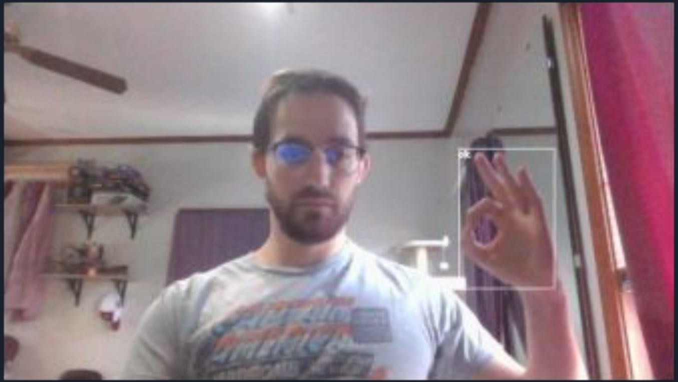
Project Description

- Hand Gesture Recognition (Like, Dislike, Okay, and Stop)
- Object detection solution
  - Identify gesture location
  - Classify gesture
- Potential use of this application includes: ASL translation and self-driving car
- Language and tools:
  - Development Environment: Google Colab
  - Implementation Format: Jupyter Notebook
  - Language: Python
  - Libraries: Pytorch

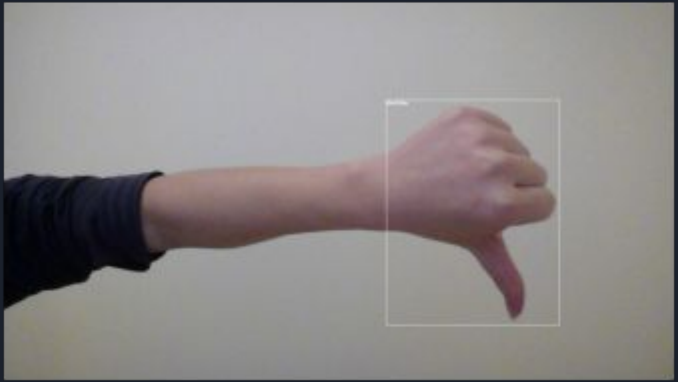
Materials

Datasets

- Several datasets made, including simple and complex variants
- Hand-crafted, including photos and annotations
  - Script to automate train/test/validation set splits
- Annotations in YOLO format, done using MakeSense.ai
  - Class, X Position, Y Position, Width, Height
  - 0 0.543505 0.523965 0.170343 0.450980
- Over 1000 images in total!

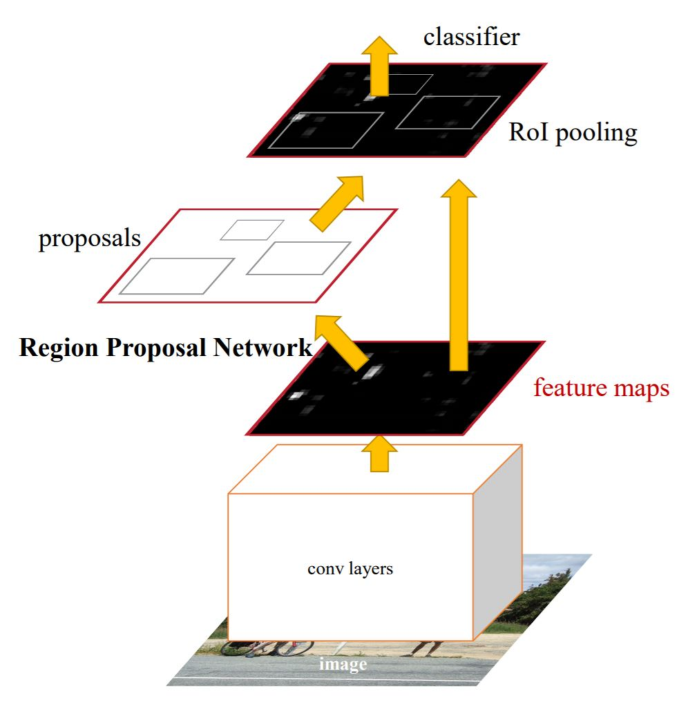


Original Dataset - Highly varied in subject and background



Simplified Dataset - Focus on gestures, simple background

Architecture

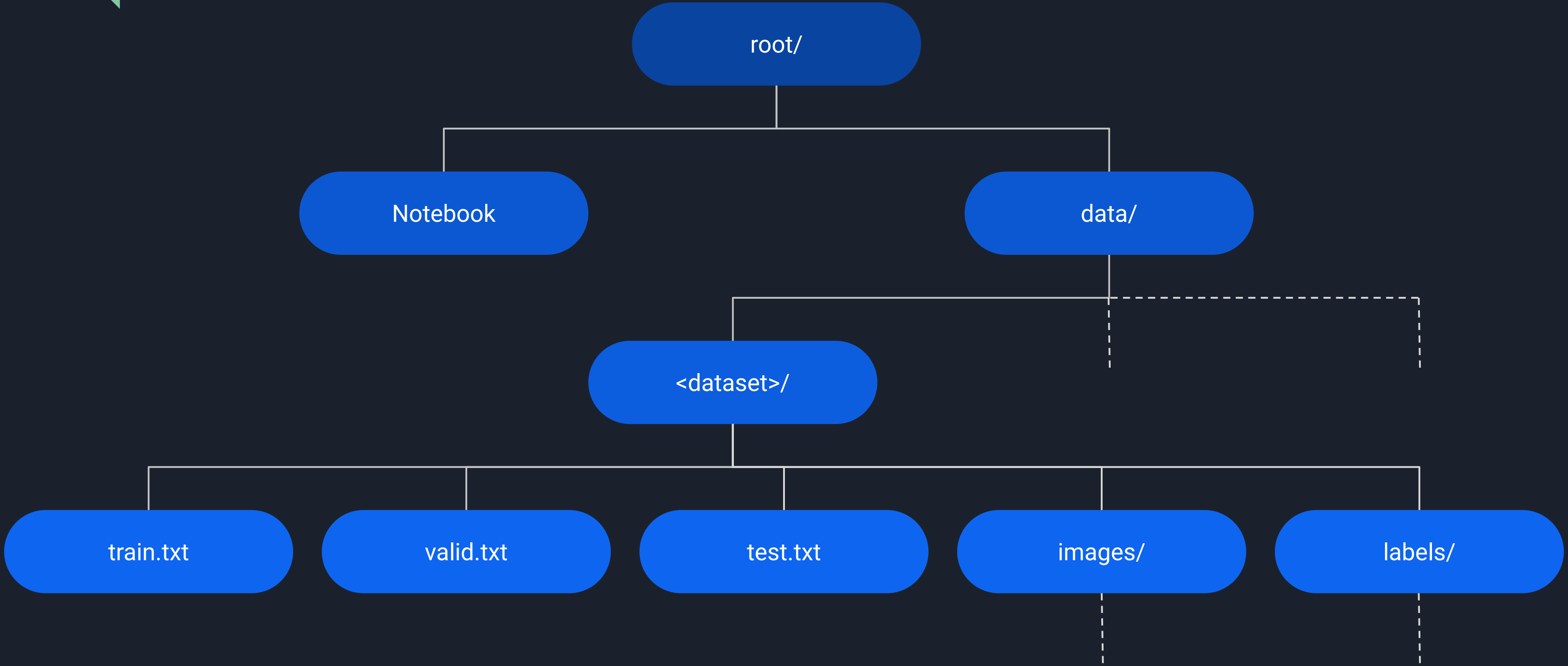


- Faster-RCNN used for final pipeline
  - ResNet 50 backbone for classification
  - RPN (Region Proposal Network) for object detection
  - Shared convolutional layers between the two
- Pre-built PyTorch Implementation
- Pretrained weights on backbone (ResNet50 classifier)
  - Weights pre-trained on ImageNet dataset

Image from Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks @ https://arxiv.org/pdf/1506.01497.pdf

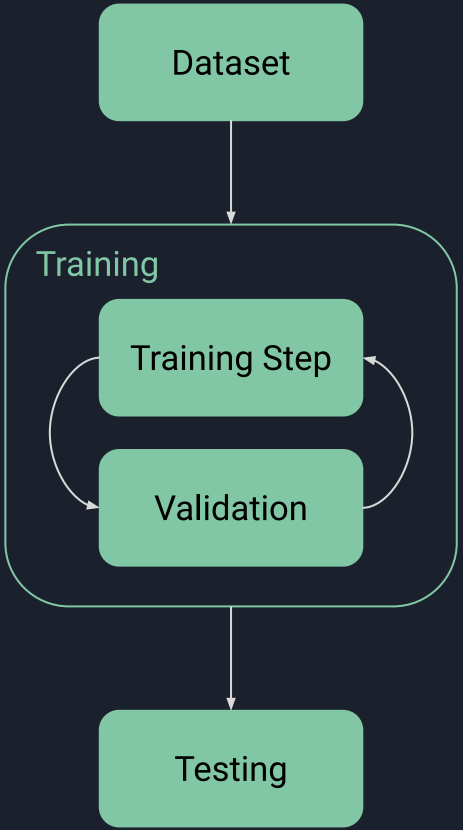
Methods

File Structure



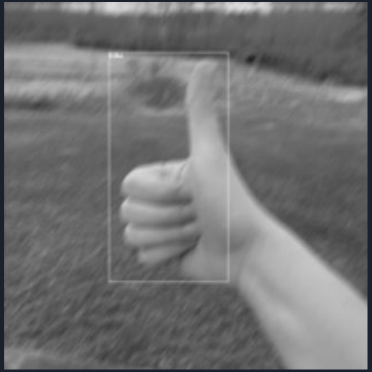
Pipeline

- Custom Dataset class to handle file structure
  - Getting items by text file, not folder structure
- Image and BBox augmentations added to increase data variance
  - Utilized Albumentations library
- Utilized Adam optimizer in training
  - Training set 70% of full dataset
  - Starting params: Learning Rate 0.0001, weight decay 0.0005
- Validate on validation set every training step
  - Validation set 10% of full dataset
  - Validate with NMS filtered classification loss
- Test
  - Test set 20% full dataset
  - Tested on NMS filtered classification accuracy, includes confusion matrix

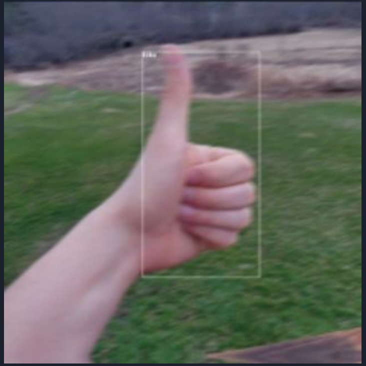


Augmentations

- Augment existing dataset with image variations using Albumentations
- Transformation pattern includes:
    - Horizontal flip
    - Median Blur
    - Rotation
    - Grayscale



Example 1: Grayscale + Blur

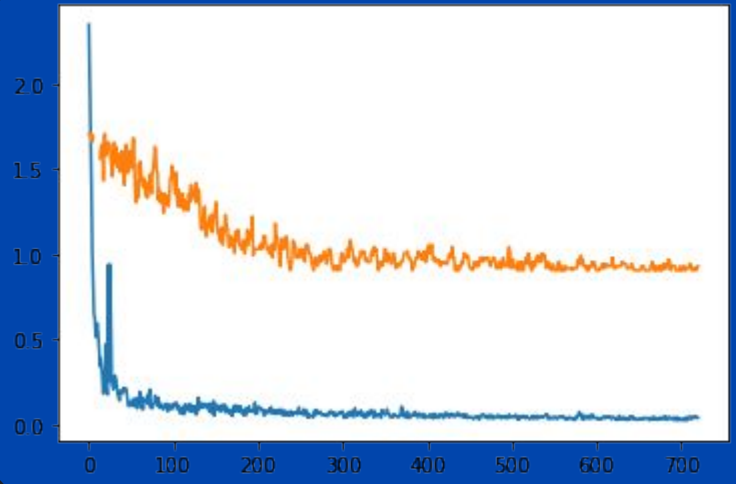


Example 2: Horizontal Flip + Blur

Validation & Testing

<u>Validation:</u>

- Validation set consists of 10% full dataset
- Validate on classification loss after NMS filtering
- Validate on all images of the validation set per training step
- Plot training and validation loss curves



<u>Testing:</u>

- Testing set consists of 20% full dataset
- Computing total and per class classification accuracy
- Producing confusion matrix

Results

Run 1: No transforms

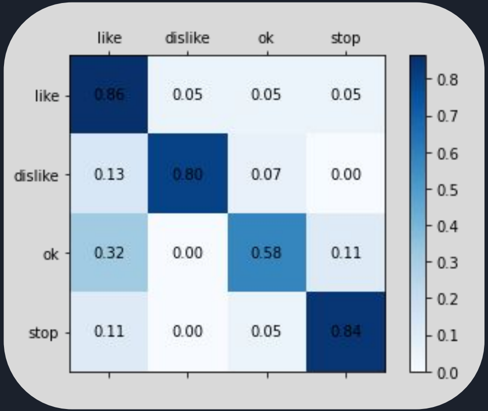
Learning Rate: 0.0001, Weight Decay: 0.0005, Batch size: 4, Epochs: 2

Basic Dataset:

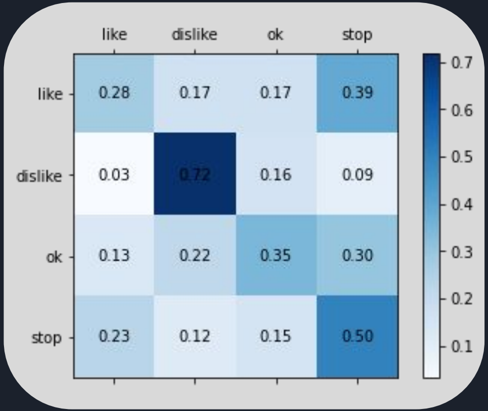
- Average IoU:
  - 0.532
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 77%

Advanced Dataset:

- Average IoU:
  - 0.4222
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 49%



	like	dislike	ok	stop
like	0.86	0.05	0.05	0.05
dislike	0.13	0.80	0.07	0.00
ok	0.32	0.00	0.58	0.11
stop	0.11	0.00	0.05	0.84



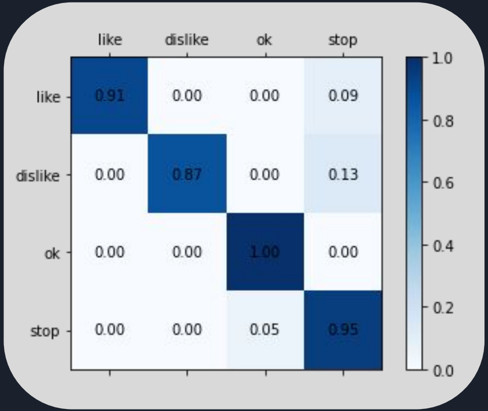
	like	dislike	ok	stop
like	0.28	0.17	0.17	0.39
dislike	0.03	0.72	0.16	0.09
ok	0.13	0.22	0.35	0.30
stop	0.23	0.12	0.15	0.50

Run 2: No transforms

Learning Rate: 0.0001, Weight Decay: 0.0005, Batch size: 4, Epochs: 10

Basic Dataset:

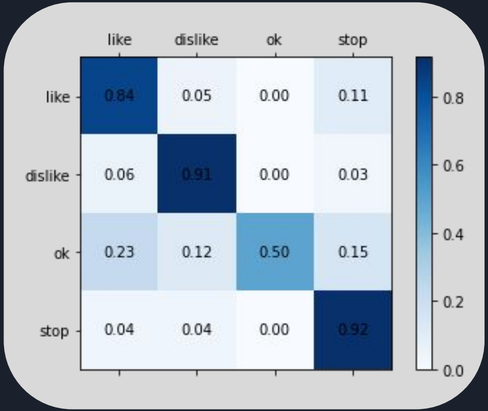
- Average IoU:
  - 0.657
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 95%



	like	dislike	ok	stop
like	0.91	0.00	0.00	0.09
dislike	0.00	0.87	0.00	0.13
ok	0.00	0.00	1.00	0.00
stop	0.00	0.00	0.05	0.95

Advanced Dataset:

- Average IoU:
  - 0.4308
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 80%



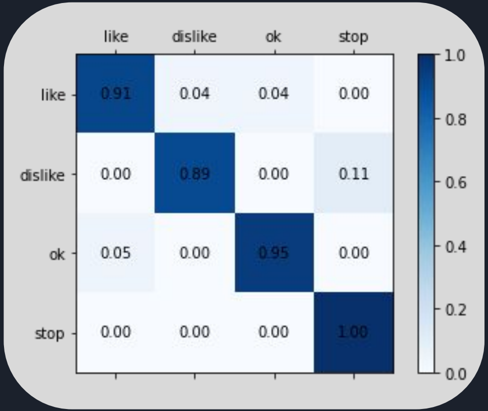
	like	dislike	ok	stop
like	0.84	0.05	0.00	0.11
dislike	0.06	0.91	0.00	0.03
ok	0.23	0.12	0.50	0.15
stop	0.04	0.04	0.00	0.92

Run 3: With Transforms

Learning Rate: 0.0001, Weight Decay: 0.0005, Batch size: 4, Epochs: 20

Basic Dataset:

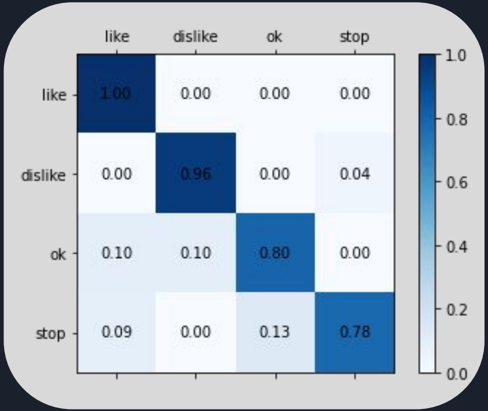
- Average IoU:
  - 0.5955
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 94%



	like	dislike	ok	stop
like	0.91	0.04	0.04	0.00
dislike	0.00	0.89	0.00	0.11
ok	0.05	0.00	0.95	0.00
stop	0.00	0.00	0.00	1.00

Advanced Dataset:

- Average IoU:
  - 0.5849
- Average Classification Accuracy given NMS filtering at 0.95 IoU
  - 88%

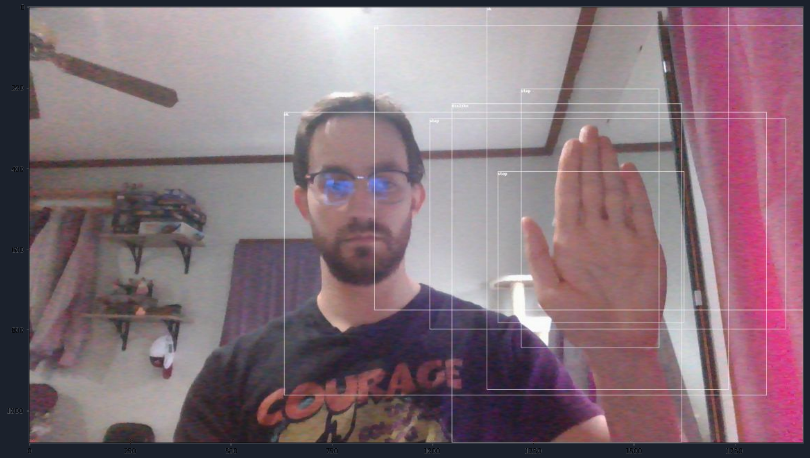


	like	dislike	ok	stop
like	1.00	0.00	0.00	0.00
dislike	0.00	0.96	0.00	0.04
ok	0.10	0.10	0.80	0.00
stop	0.09	0.00	0.13	0.78

Example Output:

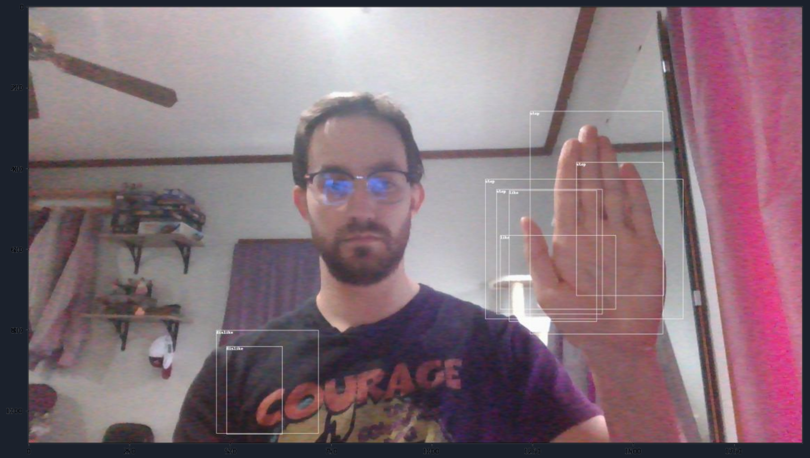
Basic dataset trained at:

LR 0.0001, Weight Decay 0.0005, 10 epochs



Advanced dataset trained at:

LR 0.0001, Weight Decay 0.0005, 10 epochs



Conclusion

- The model is capable to recognize hand gestures based on:
  - Relatively small dataset
  - Both simple and complex dataset
- High classification accuracy: 77 - 95%
- Object detection can be still improved: Average IoU 0.422 - 0.657

Discussion

<u>Difficulties:</u>

- Google Colab resource limitations
- Problems implementing and debugging original YOLOv3 implementation
  - Issues converting original pipeline to Faster-RCNN
- Learning to create good datasets from scratch
  - Annotation is time consuming
- Current augmentation pipeline fails for some data

<u>Points of Note:</u>

- NMS threshold was chosen to focus on classification accuracy
  - Reducing the threshold will give a better representation of the model’s detection abilities

Outlooks

Improvements:

- Further increase the size of our dataset
  - Experiments have shown more data increases model performance
- Redesign validation and testing pipelines to increase performance
  - Compute and add more losses, such as objectness and bbox regression
- Evaluate on more metrics, such as precision and recall
- Increase model’s detection capabilities

Avenues of Further Exploration:

- Increasing number of learned gestures
  - Potential applications: ASL translation