Object Detection for Automated Doorstep Package Classification

J. Andrus, M. Chen, C. Monis, J. Kumari

Introduction

- Why Package Detection?
  - The market has placed a high value on household monitoring technology
  - Rise in package theft

- What will this project accomplish?
  - Generate Package Classification Dataset
  - Train a YOLOv5 model
  - Export the model to run in real-time on the edge

Data loading & Preprocessing

Data Sources

Classes and Annotation

Data Augmentation

Data Sources

- Original dataset of 865 images
  - Pre-existing, publicly available package datasets
  - Frames extracted from videos captured in the test environment

Classes and Annotation

Class	Operationalization
`box`	Any rectangular cardboard package that holds its own shape rather than assuming the approximate shape of the object inside.
`plastic_bag`	Any soft plastic wrapping that does not hold its own shape and roughly assumes the shape of its contents
`envelope`	Any flat, rectangular packaging made of thick paper or thin cardboard with a single defined opening that is only designed to transport paper, thin books, or other flat objects.

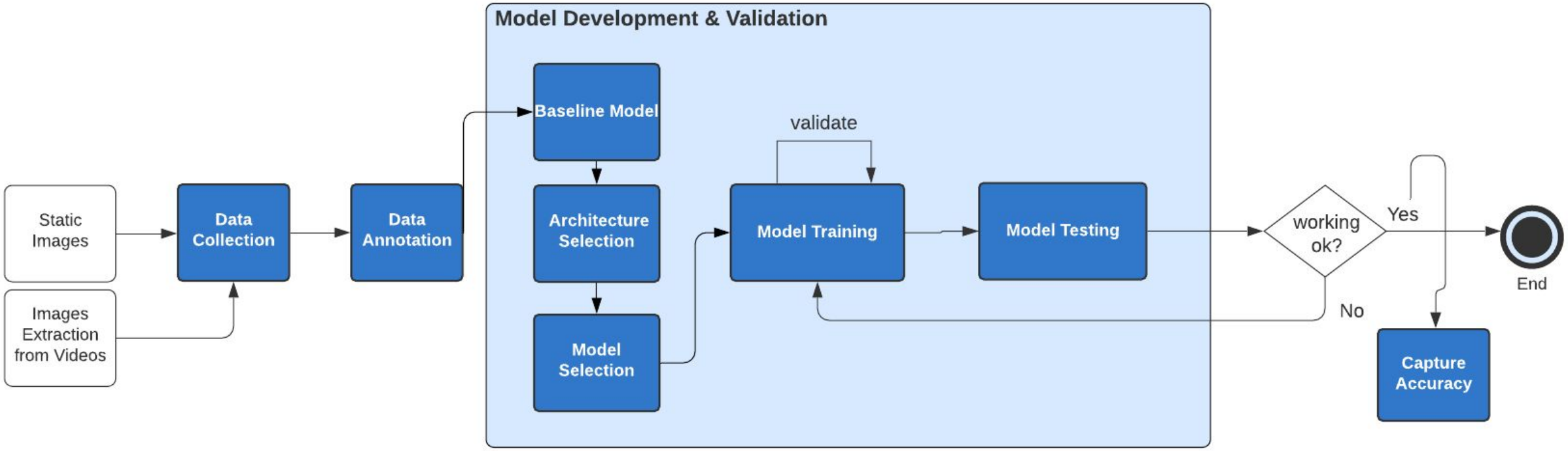
Augmentation

- Expanded dataset with a variety of transformation functions approximating weather, lighting changes, and motion
- Increased number of annotated images from 865 to 6000+

Model Training & Validation

Training & Validation Flow Diagram
Architecture Selection
Model Configuration
Detection

Model Development and Validation



Architecture Selection

- Specified number of steps/epochs required to train in <30 minutes

Architecture	mAP@0.5	mAP@0.5:0.95
MobileNetSSD	0.8338	0.5214
Faster-RCNN	0.8391	0.5305
YOLOv5	0.9515	0.6059

Model Configuration

Small
YOLOv5s

14 MB_{FP16}
2.2 ms_{V100}
36.8 mAP_{COCO}

Medium
YOLOv5m

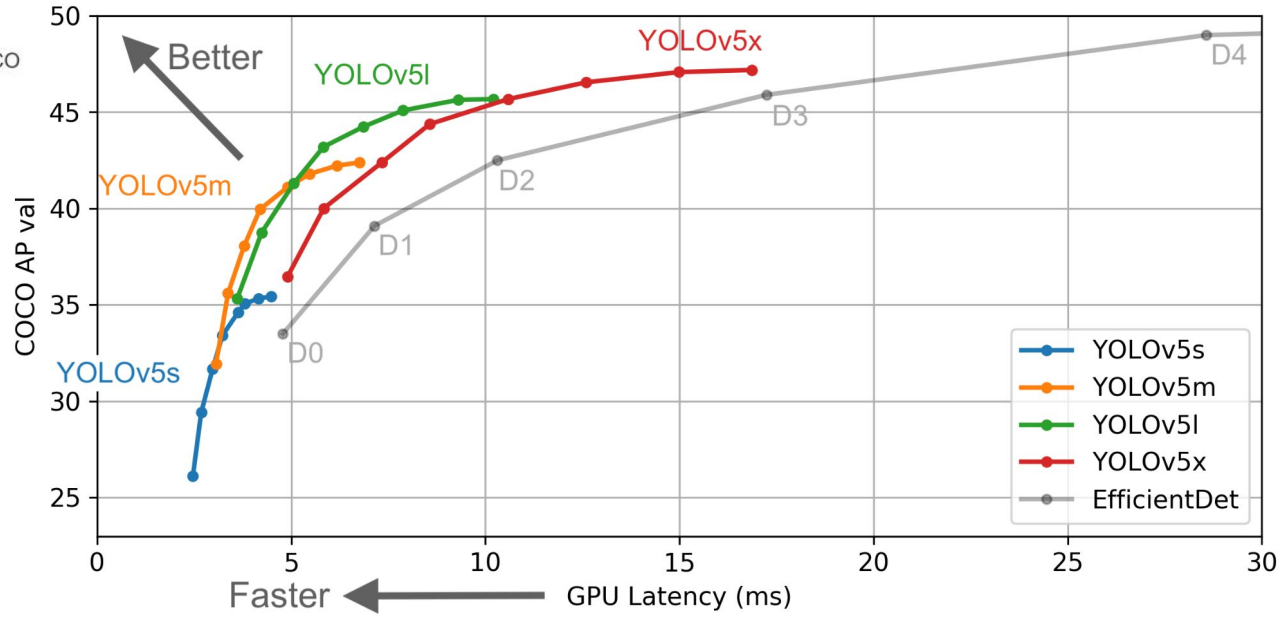
41 MB_{FP16}
2.9 ms_{V100}
44.5 mAP_{COCO}

Large
YOLOv5l

90 MB_{FP16}
3.8 ms_{V100}
48.1 mAP_{COCO}

XLarge
YOLOv5x

168 MB_{FP16}
6.0 ms_{V100}
50.1 mAP_{COCO}



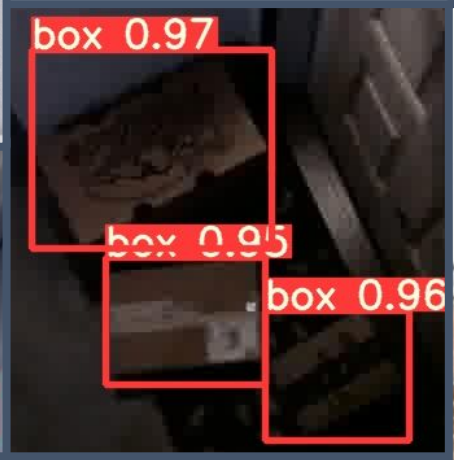
Model Selection

Configuration	Train Time (100 epochs)	mAP@0.5	mAP@0.5:0.95
yolov5s	01:22:17	0.694	0.558
yolov5x	07:20:30	0.688	0.551

Detection

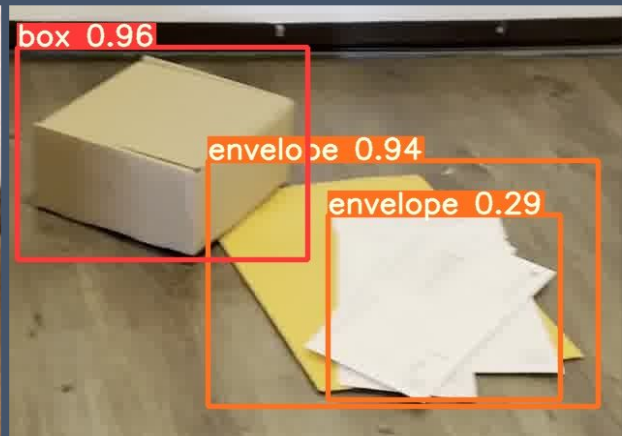

box 0.96


box 0.90
box 0.89
box 0.92
plastic_bag 0.95


box 0.97
box 0.96


box 0.95


envelope 0.95
box 0.95


box 0.96
envelope 0.94
envelope 0.29

Inference with Nx

Nx Architecture

Results

NX Architecture

Edge container applications have been deployed using Kubernetes (K3s) on the Jetson NX.

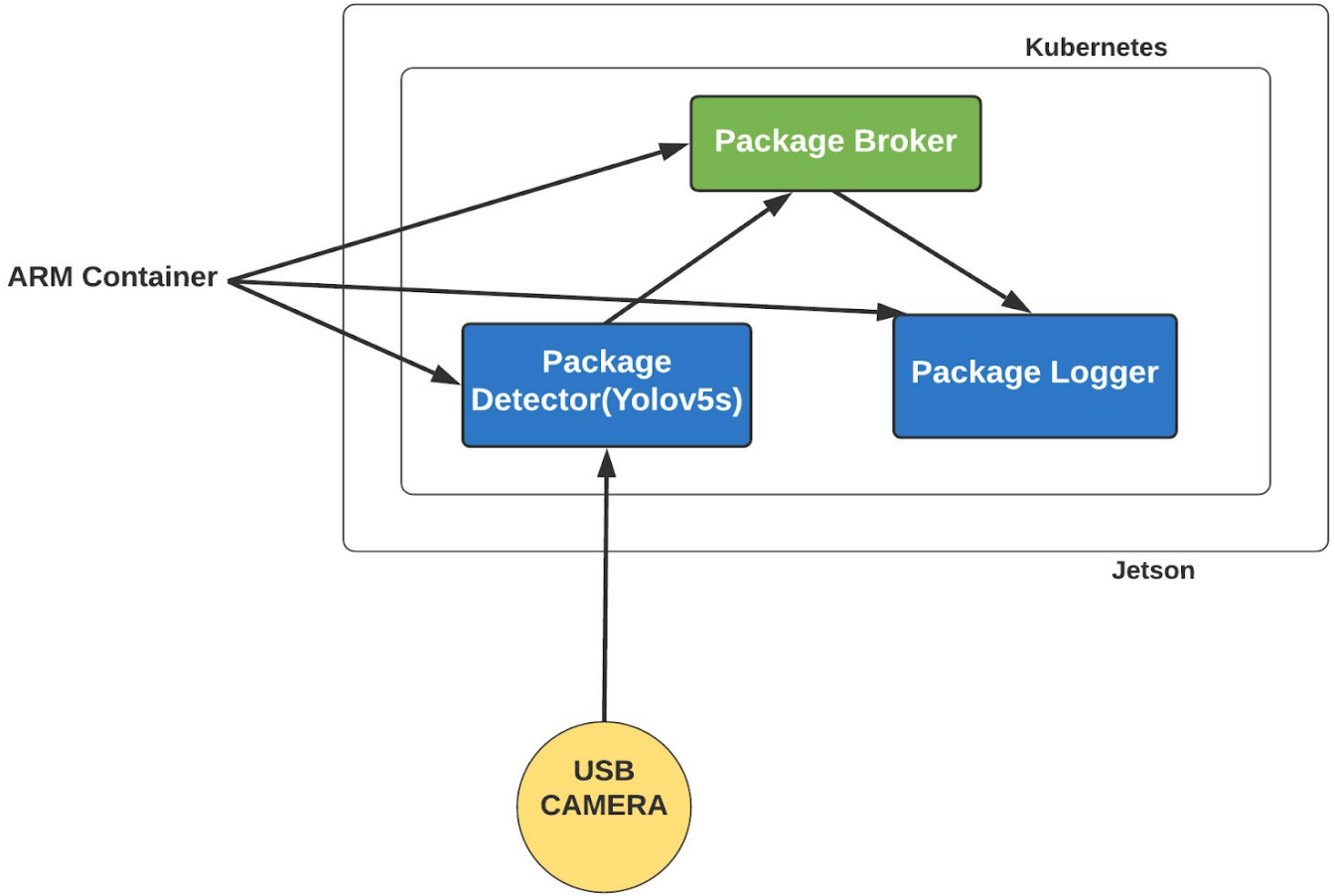
The package detector application is able to detect packages in a video stream coming from the USB camera.

Three containers:

Package Detector
- Captures video stream from USB Camera, pre-processing is done using openCV.
- Does inference with Included trained weights from custom yolov5 model.
- Publishes message to MQTT broker with detected packages and classes.

Package Broker
- Uses “MQTT” as the messaging fabric
- Includes service, to access the broker from outside Kubernetes and then from inside it.

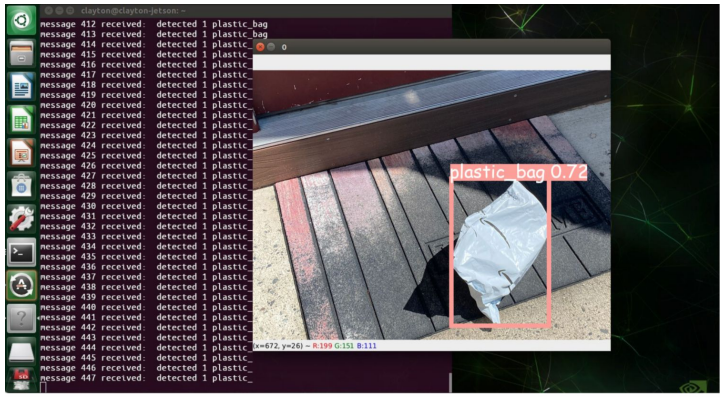
Package Logger
- Receives messages from the local broker, and outputs to its log about receiving one package or multiple packages.
- Logs the class name of the package detected.

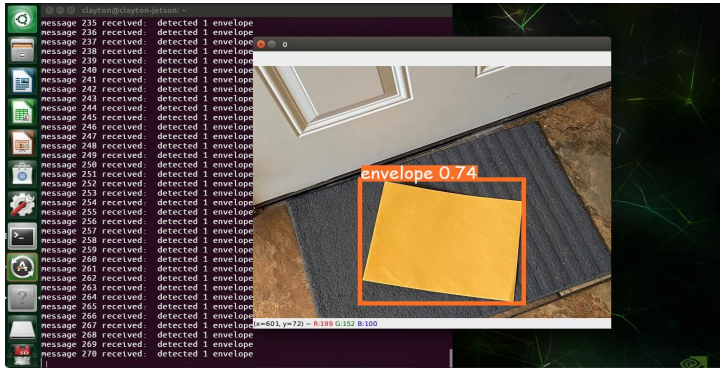


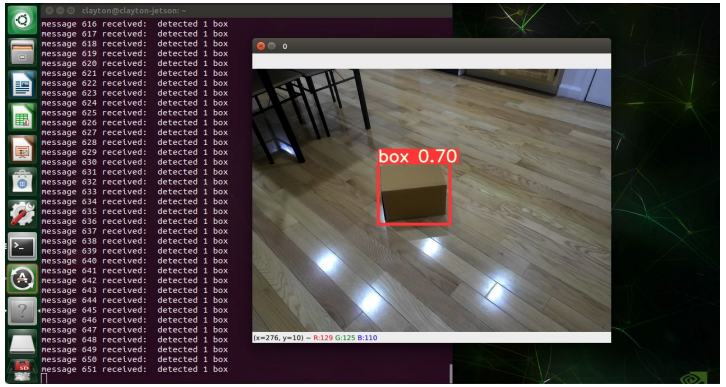
Change detect.py from Yolov5s original model as package-detector.py for running on NX

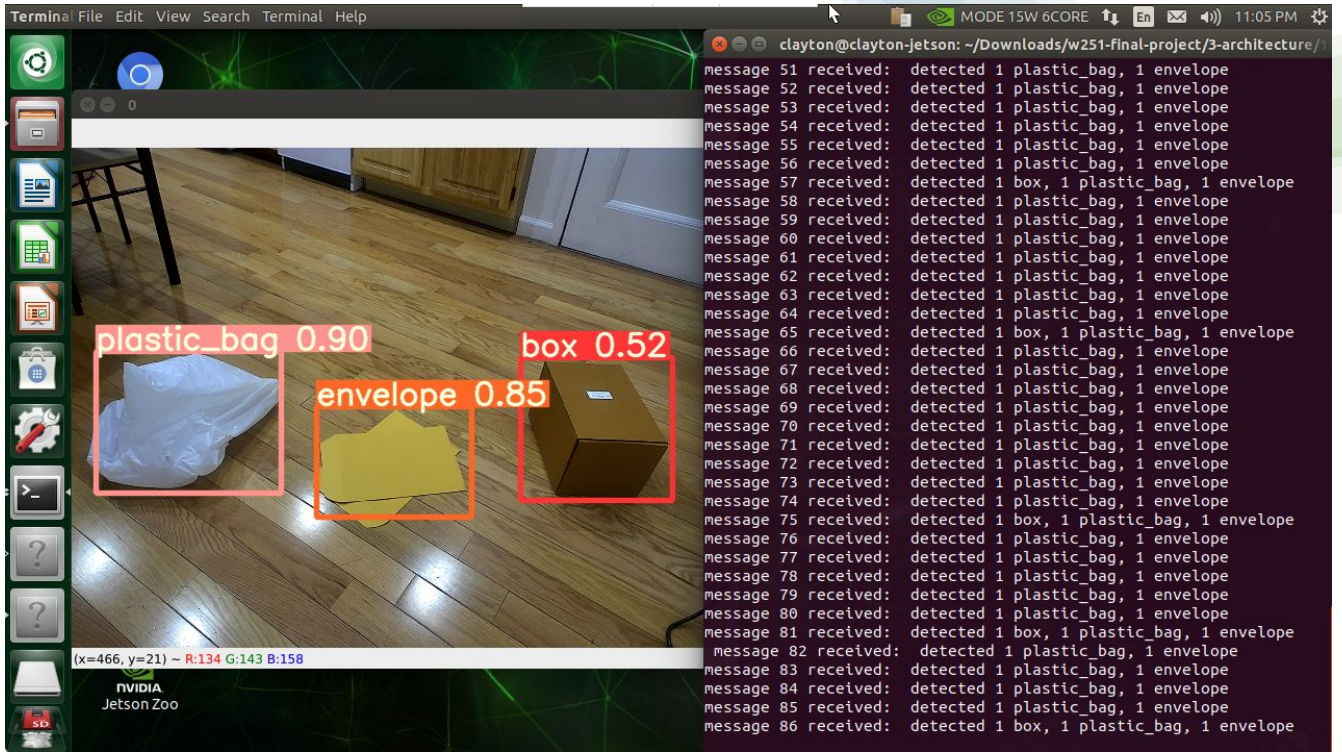
```
- Added and Initialized MQTT client to connect to local mosquitto service
      - publish the detected objects for message logger to log.
- Changed default variables of detect.py
      - changed image size to 416
      - changed default input to webcam
      - changed default confidence threshold
```

Inference


plastic_bag 0.72


envelope 0.74


box 0.70


Terminal File Edit View Search Terminal Help
MODE 15W 6CORE
11:05 PM
clayton@clayton-jetson: ~/Downloads/w251-final-project/3-architecture/
plastic_bag 0.90
envelope 0.85
box 0.52
(x=466, y=21) ~ R:134 G:143 B:158
NVIDIA
Jetson Zoo
message 51 received: detected 1 plastic_bag, 1 envelope
message 52 received: detected 1 plastic_bag, 1 envelope
message 53 received: detected 1 plastic_bag, 1 envelope
message 54 received: detected 1 plastic_bag, 1 envelope
message 55 received: detected 1 plastic_bag, 1 envelope
message 56 received: detected 1 plastic_bag, 1 envelope
message 57 received: detected 1 box, 1 plastic_bag, 1 envelope
message 58 received: detected 1 plastic_bag, 1 envelope
message 59 received: detected 1 plastic_bag, 1 envelope
message 60 received: detected 1 plastic_bag, 1 envelope
message 61 received: detected 1 plastic_bag, 1 envelope
message 62 received: detected 1 plastic_bag, 1 envelope
message 63 received: detected 1 plastic_bag, 1 envelope
message 64 received: detected 1 plastic_bag, 1 envelope
message 65 received: detected 1 box, 1 plastic_bag, 1 envelope
message 66 received: detected 1 plastic_bag, 1 envelope
message 67 received: detected 1 plastic_bag, 1 envelope
message 68 received: detected 1 plastic_bag, 1 envelope
message 69 received: detected 1 plastic_bag, 1 envelope
message 70 received: detected 1 plastic_bag, 1 envelope
message 71 received: detected 1 plastic_bag, 1 envelope
message 72 received: detected 1 plastic_bag, 1 envelope
message 73 received: detected 1 plastic_bag, 1 envelope
message 74 received: detected 1 plastic_bag, 1 envelope
message 75 received: detected 1 box, 1 plastic_bag, 1 envelope
message 76 received: detected 1 plastic_bag, 1 envelope
message 77 received: detected 1 plastic_bag, 1 envelope
message 78 received: detected 1 plastic_bag, 1 envelope
message 79 received: detected 1 plastic_bag, 1 envelope
message 80 received: detected 1 plastic_bag, 1 envelope
message 81 received: detected 1 box, 1 plastic_bag, 1 envelope
message 82 received: detected 1 plastic_bag, 1 envelope
message 83 received: detected 1 plastic_bag, 1 envelope
message 84 received: detected 1 plastic_bag, 1 envelope
message 85 received: detected 1 plastic_bag, 1 envelope
message 86 received: detected 1 box, 1 plastic_bag, 1 envelope

Conclusion

- Challenges
  - Envelope annotation

- Potential enhancements:
  - More parameter tuning
  - Delivery/pick up service support
  - Porch pirate prevention
  - Spatial Augmentations

- Summary:
  - YOLOv5s has the best performance
  - Identify multiple classes of packages in real time with average accuracy above 70%.
  - People will get real-time notifications

Reference

[1]F. Lardinois: *Hellman & Friedman deal values SimpliSafe at $1B*. Published June 29, 2018. Retrieved from https://techcrunch.com/2018/06/29/hellman-friedman-acquires-controlling-interest-in-simplisafe/.

[2]L. Stevens, D. MacMillan: *Amazon buys smart-doorbell maker Ring in deal valued at more than $1 billion*. Published February 28, 2018. Retrieved from https://www.marketwatch.com/story/amazon-buys-smart-doorbell-maker-ring-in-deal-valued-at-more-than-1-billion-2018-02-27.

[3]R. Laycock, C. Choi: *Porch pirates statistics*. Published July 31, 2021. Retrieved from https://www.finder.com/porch-pirates-statistics.

[4]J. Nelson: *Training a TensorFlow MobileNet Object Detection Model with a Custom Dataset*. Published February 9, 2020. Retrieved from https://blog.roboflow.com/training-a-tensorflow-object-detection-model-with-a-custom-dataset/.

[5]J. Nelson: *Training a TensorFlow Faster R-CNN Object Detection Model on Your Own Dataset*. Published March 11, 2020. Retrieved from https://blog.roboflow.com/training-a-tensorflow-faster-r-cnn-object-detection-model-on-your-own-dataset/.

[6]J. Nelson, J. Solawetz: *How to Train YOLOv5 On a Custom Dataset*. Published June 10, 2020. Retrieved from https://blog.roboflow.com/how-to-train-yolov5-on-a-custom-dataset/.

[7]G. Jocher: *yolov5*. Published August 1, 2021. Retrieved from https://github.com/ultralytics/yolov5.

Thank you!