Automatic Waste Detection on ZeroWaste Dataset

By Zhengqi Dong, Zeyu Gu, Tao Zhang, Yufan Lin

Main Tasks

Why

- The danger of large amount of waste production in the future.

- Demand for effectively and accurately classifying waste.

How

- Dataset:
  Zero Waste

- Method:
  - Yolov4 & YOLOR
  - Dynamic R-CNN

Related Work (Models)

YOLO series:

- One stage object detectors that have achieved state-of-the-art result.
- Fast, efficient
- We used YOLOv4 and YOLOR in our project

Dynamic R-CNN:

- Member of the R-CNN family, an example of a two-stage classifier
- Improves upon Faster R-CNN using dynamic label assignment and smooth L1 loss
- We used Dynamic R-CNN as a comparison to the one-stage detector

Related Work (Dataset)

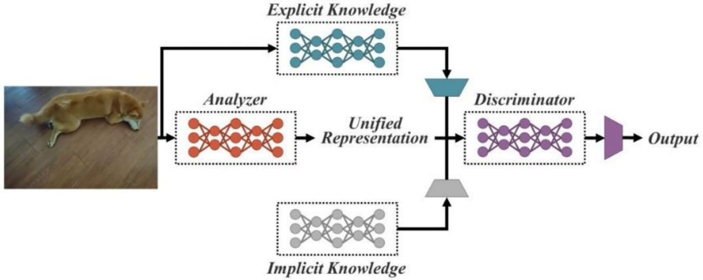
TACO:

- Open source image dataset containing waste in the wild
- Contains 1500 images with 60 classes

ReSort-IT:

- More recent dataset created for the purpose of developing better object detection models
- Contains 16000 synthetic images
- The synthetic nature is its downside

Approach 1: SIngle Stage Detector – YOLO series



- What is single stage detector? Some characteristics?
- YOLOv4(April 2020) vs Scaled YOLOv4(Nov 2020) vs YOLOR(May 2021)
- Improvement methods that works:
  - Data Augmentation: mosaic, cutout, Higher color space(Hue, Saturation), and same simple affine transformations(e.g., sacling, rotation, shearing)
  - Regularization: Dropblock regularization
- Takeaway:
  - The performance and accuracy of YOLOR is better than that of YOLOv4, Scaled YOLOv4 and lower versions. The object detection functioning is provided with feature alignment,multi tasking and prediction refinement.
  - Data Augmentation does help to improve the performance, but the effect is limited.

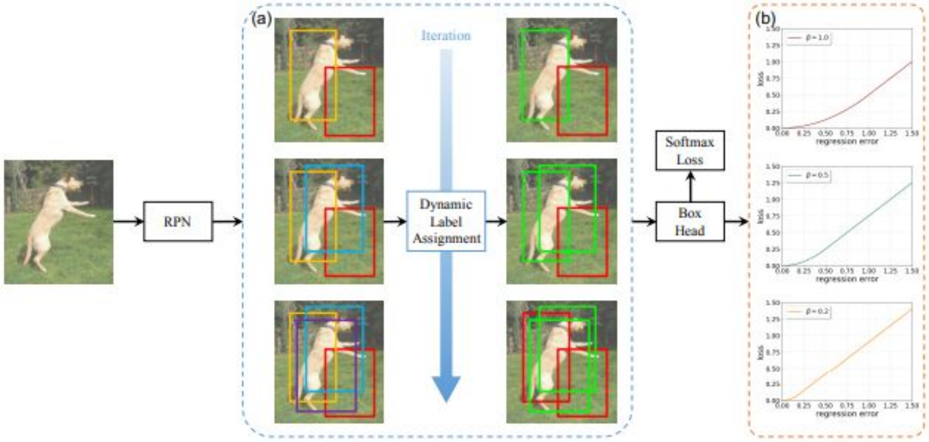
Method	mAP@.5	mAP@.5:.95
YoloV4 with no augmentation	0.27	0.164
Yolov4 with augmentation	0.406	0.26
Yolov4 with augmentation and Dropblock	0.434	0.288

Note: All result above ran 50 epochs only.

Courtesy: 1) yolor: https://arxiv.org/abs/2105.04206, 2) yolov4: https://arxiv.org/abs/2004.10934, 3) https://arxiv.org/abs/2011.08036

Approach 2: Two Stage Detector - Dynamic R-CNN

- The idea is to **dynamically change the IOU threshold** as learning improves during training to improve the performance of faster R-CNN. As more high quality proposal appears, the algorithm will increase the IOU threshold.



- Takeaway:
    - Using the **pretrained** Dynamic R_CNN model could **increase** the performance
    - Adding **regularization** to different extents **does not affect** the overall performance much

Method	mAP@.5	mAP@.5:.95
Dynamic R-CNN from scratch	0.403	0.271
Pretrained Dynamic R-CNN	0.426	0.308

Courtesy: Dynamic R-CNN, https://arxiv.org/abs/2004.06002

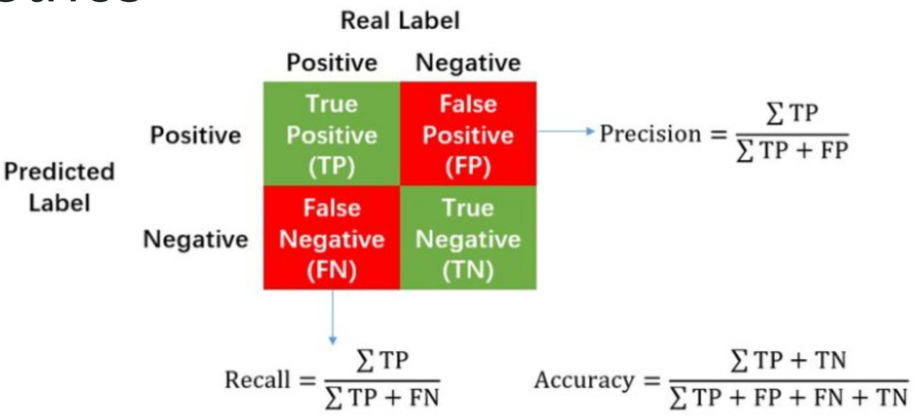
Dataset

ZeroWaste-f Dataset:

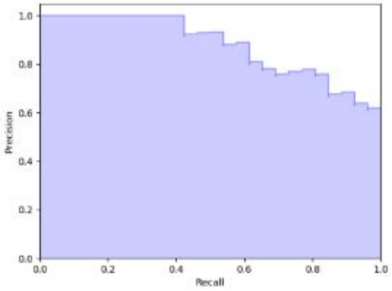
- Dataset created by professor Saenko’s research group
- Designed for industrial waste detection
- Resembles scenarios in real Material Recovery Facilities
- Images annotated by professional annotators
- Contains 4.5K images with 4 classes

Courtesy: ZeroWaste Repo, https://github.com/dbash/zerowaste

Object Detection Evaluation Metrics



Goal: For each detection, predict object’s bounding boxes, class labels, and confidence
Precision: true positive detections/ total detections
Recall: true positive detections / total positive detections in ground-truth instance
IoU (Intersection over union): Measure the amount of overlapped region in percentage
AP: For each class, sort detections from highest to lowest confidence, plot Recall-Precision curve, and compute the area under the curve.
mAP: Take average of AP over all categories. ➔ In coco, AP is mAP, and a 101-point interpolated AP definition is used in the calculation.



What we use in our project:

- AP # AP@[.5:.95]: corresponds to the average AP for IoU from 0.5 to 0.95 with a step size of 0.05 (primary challenge metric)
- $AP^{IoU=.50}$ # AP at IoU=.50 (PASCAL VOC metric)
- $AP^{IoU=.75}$ # AP at IoU=.75 (strict metric)
- AP^{smal} # AP for small objects: area < 322
- AP^{medium} # AP for medium objects: 322 < area < 962
- AP^{large} # AP for large objects: area > 962

Courtesy: 1) https://jonathan-hui.medium.com/map-mean-average-precision-for-object-detection-45c121a31173 2) https://cocodataset.org/#detection-eval

Results

Model	AP@[0.5:0.95]	AP50	AP75	APs	APm	APl
TridentNet (best result shown in [7])	24.2	36.3	26.6	4.8	10.7	26.1
Dynamic R-CNN	30.8	42.6	33.5	4.9	14.6	33.7
YoloV4	39.1	52.9	43.2	12.9	25.5	46.7
Scaled YoloV4	N/A	N/A	N/A	N/A	N/A	N/A
YOLOR	62.1	74.2	67.7	28.4	48.0	69.9

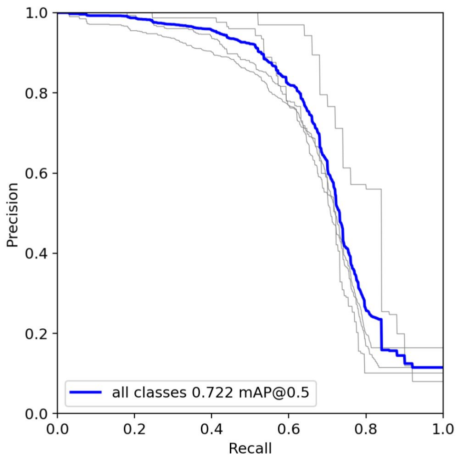


Figure: YoloR Precision-Recall curve

Note: 1) All result above were trained with 300 epochs; 2) Due to time constraints, Scaled YOLOV4 didn't get tested.

Conclusion

Till now, based upon the precision results of two approaches, Yolo R seems to be the best object detection model for zero waste dataset.

(Note: We are still working on the remaining models, the final result will be stated in the final report.)

Thanks For Listening!

Backup Slide

all	448	2.7e+03	0.747	0.642	0.867	0.766
cardboard	448	1.75e+03	0.712	0.643	0.826	0.719
metal	448	50	0.786	0.66	0.902	0.831
rigid_plastic	448	187	0.794	0.599	0.898	0.812
soft_plastic	448	713	0.697	0.665	0.841	0.702

Speed: 2552.3/0.6/2552.9 ms inference/NMS/total per 640x640 image at batch-size 32

test_batch0_labels vs test_batch0_pred(right)

test_batch1_labels vs test_batch1_pred(right)

test_batch2_labels vs test_batch2_pred(right)