Enhancing Autonomous Navigation: Few-Shot 2D to 3D Object Detection Using Depth Supervision

Ishan Sabane [1] ishancs@stanford.edu

Ishita Mangla [2] imangla@stanford.edu

Tejas Yogesh Deo [3] tejasdeo@stanford.edu

[1]Department of Electrical Engineering [2]Department of Computer Science [3]Department of Mechanical Engineering

Introduction

Models which use monocular camera images such as YOLO, SSD and Fast R-CNN have been promising for 2D object detection. However, these approaches have not been implemented for 3D object detection. Moreover, detecting novel objects is challenging as current models cannot adapt to new classes using few labelled images. Meta and transfer learning seem to be the suitable and promising approaches for solving this limitation. We explore few shot object detection on the KITTI dataset [1]

Figure 1. Kitti Images with 2D & 3D ground truth bounding boxes

Additionally, we augment the model to understand 3D bounding boxes from the learned 2D bounding boxes by using MAML Algorithm.

Background

The task of object detection involves object localization followed by object identification. Most models generate multiple bounding boxes like RPN for object localization which are filtered using NMS suppression or matched to ground truth boxes using specific cost function which avoids reassignments. Recent work leverages output hidden states to predict such bounding boxes which eliminates the intermediate region proposal step.

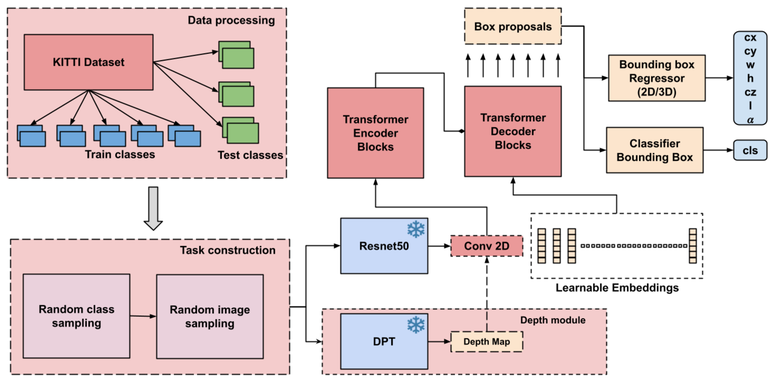

Figure 2. Combined model architecture. We build our model from top to bottom.

DETR [2] model first computes image features using a CNN backbone (ResNet-50/101) and then passes it to a Transformer with 6 Encoder and Decoder Layers, hidden dimension of 256. It passes 100 queries to the decoder which are learnable to generate 100 possible bounding boxes. To train the model, the proposed boxes are matched with the ground truth boxes using Hungarian Matching Algorithm with IOU between boxes as the cost function. Finally, the model is trained by minimizing the Regression Loss and Generalised IOU Loss for Bounding Boxes and Cross Entropy Loss for Object Labels.

Implementation Details

The Kitti Dataset consists of RBG images along with 2D and 3D bounding box coordinates and camera projection matrices for projecting 3D boxes on the 2D image. Task creation in Few-shot object detection becomes challenging due to the presence of multiple classes in a single input image. To handle this, we first pre-process the dataset and split it into respective classes without repeating images across classes. Given the large size of images, we re-scale the images and bounding boxes to match the requirements of the DETR model.

To begin with, we use extend MAML code from Home-works and implement modified version of the DETR model for Few Shot 2D Object Detection using the same loss functions as DETR.

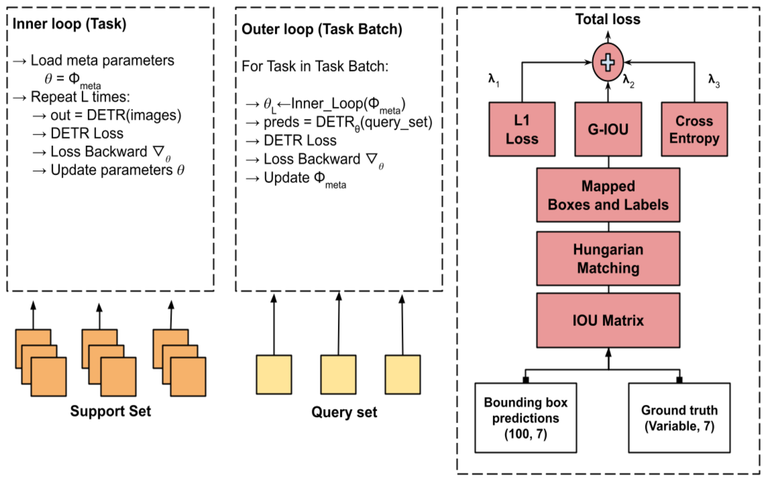

Figure 3. MAML Algorithm for 2D/3D Object Detection

To extend this to 3D bounding boxes, we modify the DETR model by adding a custom prediction head which predicts axis aligned 3D bounding box coordinates [cx,cy,w,h,cz,l,alpha] in camera coordinates(meters). We use a seperate linear head to predict 2D bounding boxes in pixels coordinates and select the first four elements from the vector for 2D bounding box. We convert the camera coordinates of 3D bounding box into pixel coordinates using camera projection matrices and take the front face of the bounding boxes for calculating IOU. We just implement 2D IoU metric by taking the smallest and largest coordinates of 3D bounding box.

Experiment Details

We train models for 2D-to-2D and 3D-to-3D few shot object detection to understand the baselines of our approach. Finally, we implement 2D-to-3D object detection and compare the results with the initial experiments. We train 3-Way 3-Shot Object detection model for all the three methods. As the training is computationally expensive, we manage to implement 5000 outer-loop steps with 50 inner-loop steps for 2D object detection. All experiments use AdamW for inner loop gradient descent with a learning rate of 1e-4 for both inner and outer loop. While we compute mAP metric over 10 IoU thresholds similar to COCO object detection, we report the mean IoU as object detection models require more steps for convergence. We also log the classification accuracy but do not optimize weight until object localization converges.

Results

To visualise the images, we first rescale the bounding box coordinates to match in input image. For 3D bounding boxes, we take the camera specific coordinates and then convert them into eight 2D coordinates and rescale them based on input image size to the DETR model.

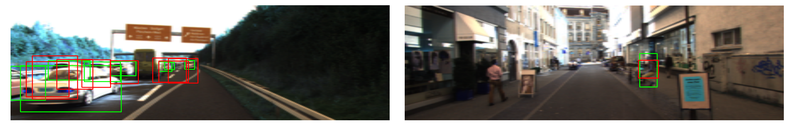

Figure 4. Visualisation of Boxes on Query Set during Meta Training and Meta Testing for 2D-to-2D task

Figure 5. Visualisation of Boxes on Query Set during Meta Training and Meta Testing for 3D-to-3D task

					Mean IoU			
					Support Set (Post Inner Loop)		Query Set (Outer Loop)	
Experiment Name	Model Name	Task Details	Inner Steps	Outer Steps	Train	Test	Train	Test
2D-to-2D	DETR-RGB	3-Way 3-Shot	50	5000	0.487	0.352	0.281	0.223
3D-to-3D	DETR-RGBD	3-Way 3-Shot	50	500	0.156	0.115	0.101	0.079
2D-to-3D	DETR-RGBD	3-Way 3-Shot	50	1000	0.346	0.296	0.003	0.001

Table 1. Few shot Object Detection Mean IoU during Meta Training and Meta Testing

Observations

- We see that few shot 2D object detection tasks adapt quickly to a new task in the meta test set. With more training, we would likely see larger improvements in object localization.
- While few shot 3D object detection seems to improve, the computational requirements for this task remain too high.
- The 2D-to-3D model adapts quickly to the 2D object detection task, but due to the shared prediction head and single outer loop optimization for 3D bounding boxes, the model parameters need to be changed to understand its performance.

References and Acknowledgements

To implement the MAML algorithm we modify the starter code from Homework 2. We also use DETR model implementation from Meta (GitHub) and update the model to allow batching and custom prediction heads.

[1] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[2] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, pp. 213–229, Springer, 2020.