# Partially Supervised Object Detection

Aartika Rai

University of Massachusetts, Amherst

## Objectives

**Partial supervision:** Full supervision for a subset of classes i.e. class and bounding box labels; weak supervision for the rest i.e. only class labels.
**Problem:** Learn to predict bounding boxes for all classes.
**Solution**: Meta learning approach. Learn to predict bounding box regression network weights instead of bounding box coordinates. Generalization in this case means being able to predict weights for new classes.

- Predict bounding box regression network weights from classification network weights by learning a transfer network.
- Transfer function is a neural network.
- A two-stream architecture from [1] for classification lets the classifier capture information about regions.

## Network Details

Modify the Faster-RCNN [2] architecture to add a multi-label classification head parallel to the region classification and regression heads.

**REGION PROPOSAL**

- Train Faster-RCNN's region proposal network (RPN) using the data from fully supervised set of classes.
- RPN provides region proposals for all classes based on their *objectness* scores.

**MULTI-LABEL CLASSIFICATION**

$f$: $r \times d$ vector representing features for each of the $r$ regions.
$\boldsymbol{s_C}$: $r \times c$ vector representing class scores for each region. For each region all class scores sum to 1.
$\boldsymbol{s_R}$: $r \times c$ vector representing region scores for each class. For reach class all region scores sum to 1.
$\boldsymbol{y_C}$: $c \times 1$ vector. A single score for each class for an image.

$$\boldsymbol{s_C} = \text{softmax}(W_c * f, \text{axis} = 0) \tag{1}$$

$$\boldsymbol{s_R} = \text{softmax}(W_r * f, \text{axis} = 1) \tag{2}$$

$$\boldsymbol{y_C} = \text{sum}(\boldsymbol{s_C} \odot \boldsymbol{s_R}, \text{axis} = 1) \tag{3}$$

**BOUNDING BOX REGRESSION**

Assuming a single layer fully connected neural network for the transfer function.
$W_{reg}$: Regression network weights.
$W_{transfer}$: Weights of the transfer network.
Compute regression weights for the $i$-th class, $W^i_{reg}$, from $W_c$ and $W_r$ as follows:

$$W^i_{reg} = \text{relu}(W_{transfer} * [W^i_c; W^i_r]) \tag{4}$$

Bounding box coordinates for the $i$-th class are computed as:

$$\boldsymbol{y_{reg}} = \text{sigmoid}(W^i_{reg} * f) \tag{5}$$

**REGION CLASSIFICATION**

Reuse $W_c$ from multi-label classification to get class scores for each region.

$$\boldsymbol{y_{cls}} = \text{softmax}(W_c * f, axis = 0) \tag{6}$$

## Architecture

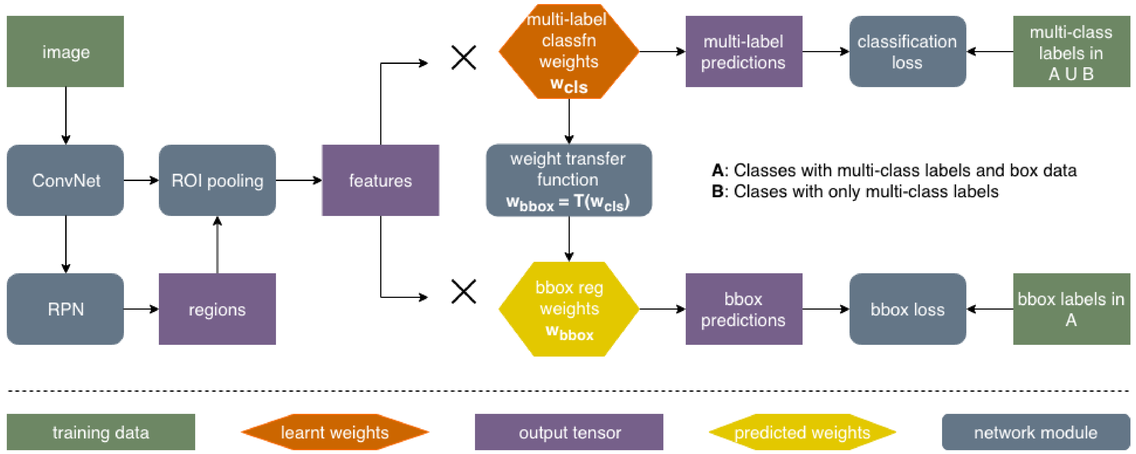


Figure 1: Model architecture showing the full pipeline with the weight transfer module $T$.

## Results

| method | aero | bike | bird | boat | bottle | bus | car | cat | chair | cow | table | dog | horse | mbike | prsn | plant | sheep | sofa | train | tv | mAP |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| exp 1 | 64.5 | 76.9 | 60 | 15.9 | 50.3 | 44.69 | 78 | 76.9 | 44.6 | 64.9 | 9.8 | 70.2 | 75.8 | 73.6 | 73.5 | 36.7 | 52.1 | 59.8 | 67.6 | 70.3 | 58.3 |
| exp 2 | 23.7 | 75.2 | 62.6 | 47.8 | 54.1 | 70.7 | 78.3 | 46.6 | 49 | 73.7 | 62 | 70.7 | 33.4 | 48.5 | 71.8 | 41.4 | 68.4 | 62.5 | 27.9 | 69.9 | 56.9 |
| F-RCNN | 70.4 | 79 | 70.2 | 56.5 | 54.2 | 78.3 | 84.4 | 84.2 | 48.2 | 78 | 63.7 | 80.8 | 84 | 75.8 | 77.5 | 47.2 | 73.4 | 67.2 | 77.3 | 73 | 71.2 |

Table 1: mAP results for the two experiments and faster-RCNN trained on full data.

## Qualitative Results

Figure 2: Experiment 1 (top) and Experiment 2 (bottom) results.

## Data

- Pascal VOC 2007: **9,963** images, **24,640** annotated objects, *5011 trainval* and *4952 test*.
- Multi-class labels are derived from bounding box data and added as an additional target.
- Partition categories into two subsets. **A:** full supervision and **B:** weak supervision.
- Bounding boxes are removed for all images that contain any object from categories in B from the training data.

## Experiments

- Experiment 1 with 5 held out classes: *boat, bus, cow, dining-table, sheep*. Set A: **4210** images, Set B: **801** images.
- Experiment 2 with 5 held out classes: *aero, cat, horse, mbike, train*. Set A: **3644** images, Set B: **1367** images.
- Transfer function is a fully connected layer with relu activation.
- VGG backbone with weights pre-trained on ImageNet.
- A dropout layer after the fully connected layer of transfer function produces better results. Results shown for dropout 0.5.

## Analysis

Some interesting trends and shortcomings:

- Precision is better for classes that have a similar class included in set A. Eg. *sheep* has *dog* and *cats* as it's counterparts, *cow* has *horse*.
- For unique categories in B such as *boat*, bounding boxes for special parts are also prediced as the it's unsure what exactly constitutes an object of this category. Eg. sail and bow for boats are also labelled.
- *Dining table* is a particularly hard category for this task as those are severly occluded in most images and almost always appear alongside other items such as plates and chairs making it harder to separate it's identity.

## References

[1] H. Bilen and A. Vedaldi.
Weakly supervised deep detection networks.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun.
Faster r-cnn: Towards real-time object detection with region proposal networks.
In *Neural Information Processing Systems*, 2015.

[3] Ronghang Hu, Piotr DollÃąr, Kaiming He, Trevor Darrell, and Ross Girshick.
Learning to segment every thing.
In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.