# Non - Maximum Suppression (NMS)

**Input** : $\mathcal{B} = \{b_1, .., b_N\}, \mathcal{S} = \{s_1, .., s_N\}, N_t$

$\qquad\qquad \mathcal{B}$ is the list of initial detection boxes

$\qquad\qquad \mathcal{S}$ contains corresponding detection scores

$\qquad\qquad N_t$ is the NMS threshold

**begin**

$\qquad \mathcal{D} \leftarrow \{\}$

$\qquad$ **while** $\mathcal{B} \neq empty$ **do**

$\qquad\qquad m \leftarrow \text{argmax } \mathcal{S}$

$\qquad\qquad \mathcal{M} \leftarrow b_m$

$\qquad\qquad \mathcal{D} \leftarrow \mathcal{D} \bigcup \mathcal{M}; \mathcal{B} \leftarrow \mathcal{B} - \mathcal{M}$

$\qquad\qquad$ **for** $b_i \ in \ \mathcal{B}$ **do**

$\qquad\qquad\qquad$ **if** $iou(\mathcal{M}, b_i) \geq N_t$ **then**

$\qquad\qquad\qquad\qquad \mathcal{B} \leftarrow \mathcal{B} - b_i; \mathcal{S} \leftarrow \mathcal{S} - s_i$

$\qquad\qquad\qquad$ **end** NMS

$\qquad\qquad\qquad s_i \leftarrow s_i f(iou(\mathcal{M}, b_i))$ Soft-NMS

$\qquad\qquad$ **end**

$\qquad$ **end**

$\qquad$ **return** $\mathcal{D}, \mathcal{S}$

**end**

Figure 2. The pseudo code in red is replaced with the one in green in Soft-NMS. We propose to revise the detection scores by scaling them as a linear or Gaussian function of overlap.

# Evaluation

## 1. Detection Evaluation

This page describes the *detection evaluation metrics* used by COCO. The evaluation code provided here can be used to obtain results on the publicly available COCO validation set. It computes multiple metrics described below. To obtain results on the COCO test set, for which ground-truth annotations are hidden, generated results must be uploaded to the evaluation server. The exact same evaluation code, described below, is used to evaluate results on the test set.

## 2. Metrics

The following 12 metrics are used for characterizing the performance of an object detector on COCO:

```
Average Precision (AP):
  AP                  % AP at IoU=.50:.05:.95 (primary challenge metric)
  AP^IoU=.50          % AP at IoU=.50 (PASCAL VOC metric)
  AP^IoU=.75          % AP at IoU=.75 (strict metric)
AP Across Scales:
  AP^small            % AP for small objects: area < 32^2
  AP^medium           % AP for medium objects: 32^2 < area < 96^2
  AP^large            % AP for large objects: area > 96^2
Average Recall (AR):
  AR^max=1            % AR given 1 detection per image
  AR^max=10           % AR given 10 detections per image
  AR^max=100          % AR given 100 detections per image
AR Across Scales:
  AR^small            % AR for small objects: area < 32^2
  AR^medium           % AR for medium objects: 32^2 < area < 96^2
  AR^large            % AR for large objects: area > 96^2
```

1. Unless otherwise specified, *AP and AR are averaged over multiple Intersection over Union (IoU) values*. Specifically we use 10 IoU thresholds of .50:.05:.95. This is a break from tradition, where AP is computed at a single IoU of .50 (which corresponds to our metric $AP^{IoU=.50}$). Averaging over IoUs rewards detectors with better localization.
2. AP is averaged over all categories. Traditionally, this is called "mean average precision" (mAP). We make no distinction between AP and mAP (and likewise AR and mAR) and assume the difference is clear from context.
3. AP (averaged across all 10 IoU thresholds and all 80 categories) will determine the challenge winner. This should be considered the single most important metric when considering performance on COCO.
4. In COCO, there are more small objects than large objects. Specifically: approximately 41% of objects are small (area < $32^2$), 34% are medium ($32^2$ < area < $96^2$), and 24% are large (area > $96^2$). Area is measured as the number of pixels in the segmentation mask.
5. AR is the maximum recall given a fixed number of detections per image, averaged over categories and IoUs. AR is related to the metric of the same name used in proposal evaluation but is computed on a per-category basis.
6. All metrics are computed allowing for at most 100 top-scoring detections per image (across all categories).
7. The evaluation metrics for detection with bounding boxes and segmentation masks are identical in all respects except for the IoU computation (which is performed over boxes or masks, respectively).