Union-over-Intersections: Object Detection beyond Winner-Takes-All

Aritra Bhowmik, Pascal Mettes, Martin R. Oswald, Cees G. M. Snoek

Rethinking Regression: Collaboration not Competition

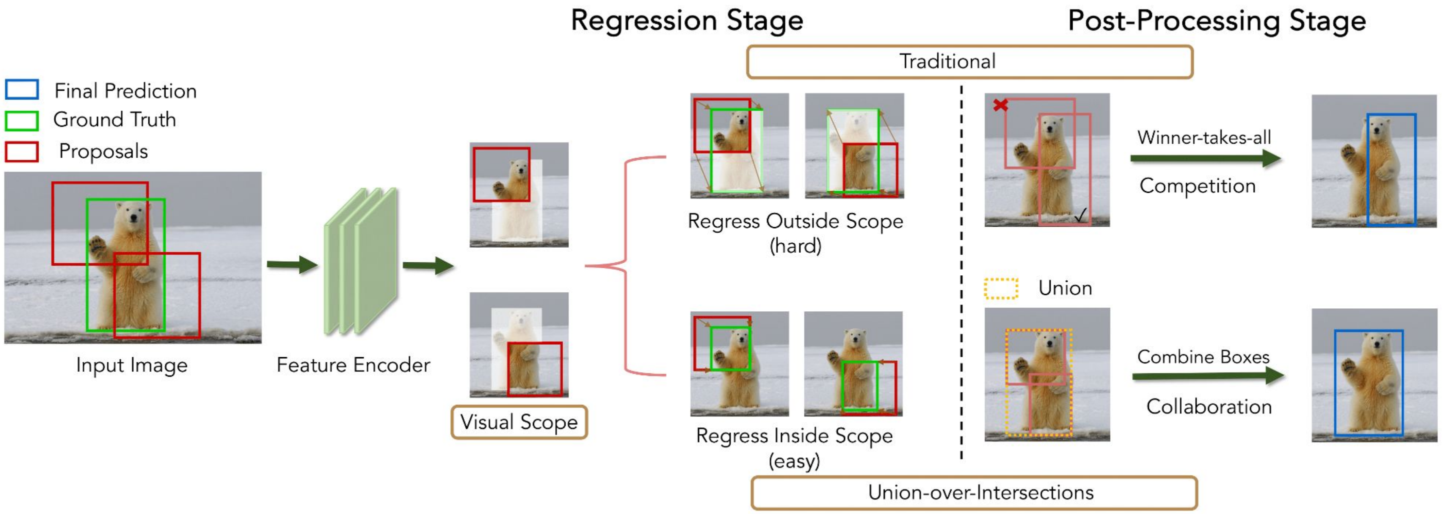
Classical Regression

- 🎯 Same target
- ❌ Out-of-scope
- 🥇 Pick one

Union-over-Intersections

- 🎯 Different target
- ✅ Within scope
- 🤝 Union of intersections

Standard vs Our Pipeline



STAGE 1
Regressing to Intersections

Easier task

STAGE 2
Union-over-Intersections

Collaboration Better

Proposed changes at a glance

Regression Stage

Input :

Set of Proposals $P = \{p_1, ..., p_N\}$
Corresponding ground truths $G = \{g_1, ..., g_N\}$
Corresponning detection scores $S = \{s_1, ..., s_N\}$
Iou threshold k

Loss and Targets for Regression :

Loss = $|T - B|$,
where $B = \{b_1, ..., b_N\}$ are the set of box predictions.

$T = \{g_1, ..., g_N\}$ **Traditional** targets

$T = \{p_1 \cap g_1, ..., p_N \cap g_N\}$ **Our** targets

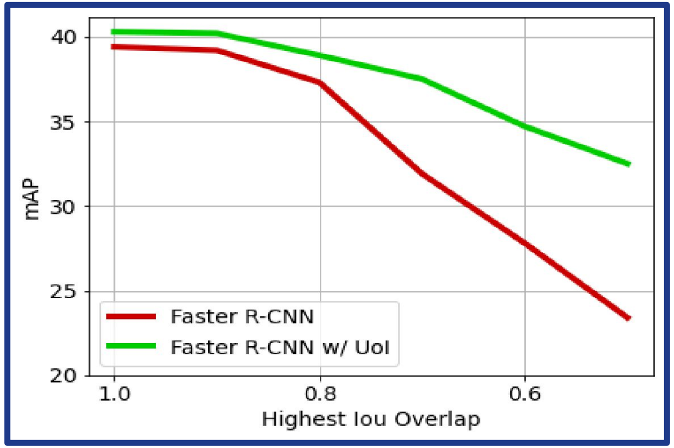
Main Difference: *We regress only to the intersections, not the whole ground truth.*

Post-Processing Stage

begin
$D \leftarrow \{\}$
while $P \neq$ empty **do**
$m \leftarrow$ argmax S
$M \leftarrow b_m$; $H \leftarrow P_m$; $C \leftarrow \{H\}$
$B \leftarrow B \setminus M$; $P \leftarrow P \setminus H$
for i in range (len(P)) **do**
if iou(M, b_i) $\geq k$ **then**
$B \leftarrow B \setminus b_i$; $S \leftarrow S \setminus s_i$
end
if iou(H, p_i) $\geq k$ **then**
$P \leftarrow P \setminus p_i$; $S \leftarrow S \setminus s_i$; $C \leftarrow C \cup b_i$
M = top 5 (C) combined
end
Main Difference: *Union over discarding.*
end
$D \leftarrow D \cup M$
end
return D, S
end

Observations

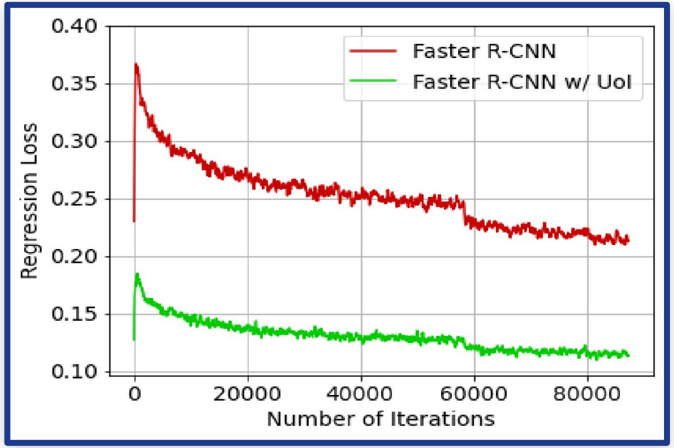
(a) Robust to bad proposals



Only using bad proposals

Performance drops : much less

(b) Simpler target

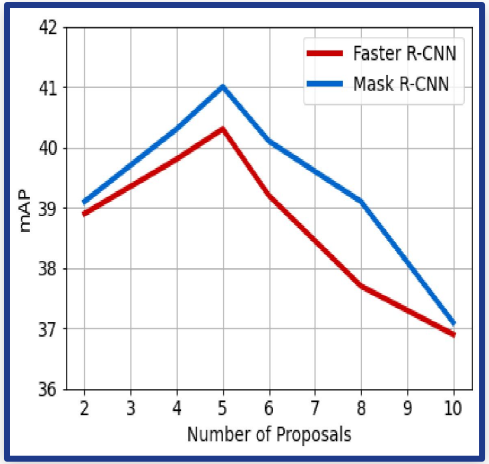


Regressing to Intersections

Half the loss at convergence

Observations

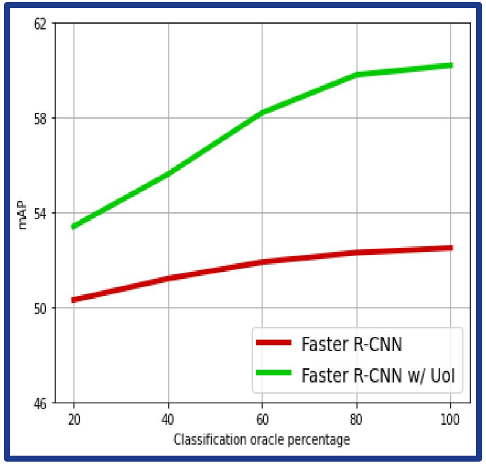
(c) Top-K proposal



Depends on proposal number

5 is the best

(d) ↑classifier, ↑improvement



Classification accuracy improves

Higher relative improvement

Detection comparison on MS COCO

Method	Backbone	mAP	AP_{50}	AP_{75}
Faster RCNN	R-50-fpn	37.4	58.1	40.4
Faster RCNN w/ UoI	R-50-fpn	**38.1**	**58.7**	**40.9**
YOLOv3	DarkNet-53	33.7	56.6	35.3
YOLOv3 w/ UoI	DarkNet-53	**34.5**	**57.5**	**35.9**
Def-DETR	R-50-fpn	44.3	63.2	48.6
Def-DETR w/ UoI	R-50-fpn	**44.8**	**63.9**	**49.1**

UoI works like a plug-and-play across architectures

Segmentation comparison on MS COCO

Method	Backbone	mAP	AP_S	AP_M
Mask RCNN	R-50-fpn	34.7	15.8	36.9
Mask RCNN w/ UoI	R-50-fpn	**35.3**	**16.2**	**37.5**

UoI works beyond detection task and improves segmentation

IoU-type loss based comparison

IoU Loss Type	Base (mAP)	Base w/ *UoI* (mAP)	Base (AP_{75})	Base w/ *UoI* (AP_{75})
L1	37.4	**38.1**	40.4	**40.9**
G-IoU	38.0	**38.6**	41.1	**42.0**
Alpha-IoU	38.9	**39.4**	41.8	**42.6**

Our **UoI** approach does not depend upon the regression loss type

Grouping method based comparison

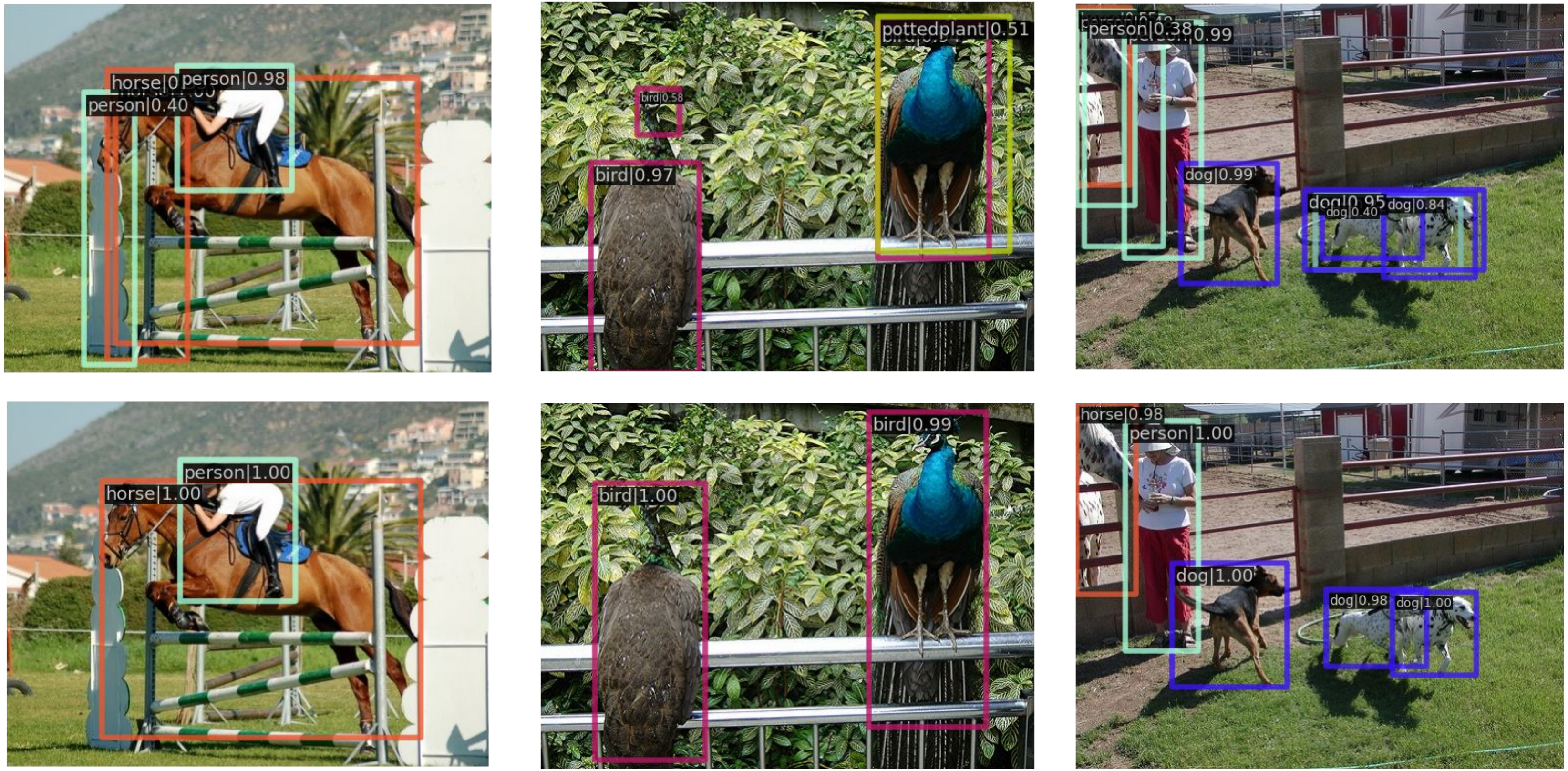
NMS Type	**Base (mAP)**	**Base w/ *UoI* (mAP)**	**Base (AP_{75})**	**Base w/ *UoI* (AP_{75})**
NMS	37.4	**38.1**	40.4	**40.9**
Cluster-NMS	37.6	**38.4**	40.4	**41.0**
Soft-NMS	38.2	**38.8**	40.9	**41.7**

UoI performed with better grouping strategies makes it better

Qualitative results

w/o UoI

w/ UoI



Corrects the stretch of the boxes via **UoI**

Failure

w/o UoI

w/ UoI

UoI sometimes merges multiple instances into one in crowded cases

THANK YOU