We thank all the reviewers for their valuable feedback. To summarize, we received three "Good Paper; accept" and a "Marginally above the acceptance threshold" ratings, which confirm the significance of our work for the community. For the rebuttal, we incorporated aLRP Loss into the “mmdetection framework", which includes the implementations of all major detectors, performed additional experiments with an anchor-free detector (FoveaBox [A]) and a label-assignment method (ATSS [B]), and started more experiments with other methods whose results will be included in the final paper.

R1.1: “*The experiment is only done for one method...*” **Authors:** We only tested aLRP Loss on RetinaNet following recent similar works (e.g. FreeAnchor [30], DR Loss [22], AP Loss [7]). Table A1 presents results with FoveaBox [A], a state-of-the-art anchor-free detector (similar to FCOS), by replacing Focal Loss and SmoothL1 with aLRP Loss. We observed that aLRP Loss (i) improves FoveaBox and (ii) discards four hyperparameters of the default loss function. However, these results are not final because we used RetinaNet’s optimal learning rate and schedule, which we will tune for FoveaBox. Our experiments are still in progress for two-stage detectors and other one-stage detectors.

R1.2: “*...I wonder how these label-assign strategies work with the aLRP loss. Can they be further improved?*” **Authors:** Table A2 shows that aLRP Loss and ATSS [B] are complementary. aLRP Loss improves ATSS by around 1 AP. Furthermore, we notice that ATSS decreases training time of aLRP Loss by using fewer anchors (i.e. positives). We thank R1 for this contribution. We will provide a discussion on the combination of ATSS and aLRPLoss in the final version.

R1.3: “*... the unified confidence for each anchor should have been modeling in FreeAnchor.*” **Authors:** Agreed. We will move FreeAnchor under “Methods Combining Branches” in Table 4.

R2.1: “*The performance of object detection at* 512×512 *resolution is lower ... It will be more interesting if the performance can outperform the state-of-the-arts.*” **Authors:** aLRP Loss does outperform all methods with the same backbone and similar test scales. For example, for 500 scale, we improve the closest counterpart, HSD [2], by 1.5 AP. Please see lines 218-220 and Table 4.

R2.2: “ *the general applicability of aLRP to object detectors ... was not valided.*” **Authors:** Please see **R1.1** and **R1.2**.

R2.3: “*... can the aLPR approach works given test images of very sparse objects.*” **Authors:** Table A3 compares the performance of methods on COCO minival for different number of objects per image. aLRPLoss has a significant gain (i.e. ~ 4.5 AP) for very sparse images (i.e. 0-3 interval), and outperforms other methods in each sparsity level.

R3.1: “*... I encouge the authors to try the loss on more detectors...*” **Authors:** Please see **R1.1** and **R1.2**.

R4.1: “*... An ablation study introducing an additional loss weight hyper-parameter or learned loss weight like [11] will be helpful.*” **Authors:** As we wrote in Section 4.2, we use a self-balancing (SB) strategy between localisation and classification, which is hyper-parameter free, simple and theoretically validated. Table A4 compares our SB method with scalar weighting. The results suggest that SB discards tuning w_r, and slightly yields better performance. We thank R4 for pointing out this and will include this ablation experiment in the paper.

R4.2: “*...not clear to me the intuition behind using the classifier branch output (rank (i)) for the regression task.*” **Authors:** A large rank for a positive example implies a lower score (s_i), therefore including rank(i) in the denominator decreases its contribution to the localisation loss, which focuses the training of the localisation branch on positive boxes with larger scores (i.e. those with smaller rank(i)). Fig.2(c) and Lines 146-153 discuss this interaction. We will make it more clear. Also note that rank(i) is the direct outcome of converting LRP to loss term $\ell^{\mathrm{LRP}}(i)$ and hence ensures comparable ranges for classification and localisation components.

R4.3: “*...I strongly advise showing the effectiveness of the proposed method on other object detectors. Is the method applicable to two-stage object detectors and recent anchor-free methods...*” **Authors:** Please see **R1.1** and **R1.2**.

[A] Kong et al., “FoveaBox: Beyond Anchor-based Object Detector”, IEEE Transactions on Image Processing, 2020.
[B] Zhang et al., “Bridging the Gap Between Anchor-based and Anchor-free Detection...”, CVPR, 2020.

Table A1: Using aLRP Loss with FoveaBox [A].

Method	Epochs		
	50	75	100
Focal L.+SL1	29.5	38.5	39.8
aLRP L.	**34.0**	**39.7**	**40.3**

Table A2: Using aLRP Loss with ATSS [B].

Method	AP
ATSS	30.9
ATSS + aLRP L.	**32.0**

Table A3: Effect of Sparsity.

Method	# of objects/image			
	0-3	4-10	11-20	21-62
Focal L.+SL1	52.5	39.3	31.5	22.9
AP L.+SL1	51.9	38.3	30.4	23.9
aLRP L.	**56.3**	**42.0**	**33.5**	**25.5**

Table A4: Ablation analysis for additional scalar localisation task weight, w_r. SB: Self-Balance (see Sec. 4.2)

w_r	1	5	10	15	SB
AP	27.3	30.5	30.8	30.4	**30.9**

The models in Tables A2 and A4 are trained on COCO minitrain (https://github.com/giddyyupp/coco-minitrain), approximately a quarter of the coco-trainval. All models use ResNet-50 backbone, follow the training details for 500 scale described in Section 5 and are tested on COCO minival.