Supplementary: Do We Still Need Non-Maximum Suppression? Accurate Confidence Estimates and Implicit Duplication Modeling with IoU-Aware Calibration

Johannes Gilg Torben Teepe Fabian Herzog Philipp Wolters Gerhard Rigoll
Technical University Munich

1. Influence of t_{IoU} on IoU-aware calibration

For fitting the proposed IoU-aware calibration we need to evaluate the detections of a object detector. The concept of a True Positive (TP) for detectors is more involved than for *e.g.* classification, as it depends on the chosen t_{IoU} which defines the minimum overlap required for a detection with an actual object to be considered a TP detection. As mentioned in the Background section, in our experiments we followed Küppers *et al.* [7] and use a IoU threshold t_{IoU} of 0.5. The same threshold is used for fitting the IoU-aware calibration and for evaluating how well the detections are calibrated. The choice of t_{IoU} can impact the performance changes of conditional confidence calibrations [6].

Impact on performance. In Tab. 1 we can see the impact of t_{IoU} on the performance metrics. A t_{IoU} of 0.5 makes the concept of a TP the same for the calibration objective as it is for the evaluation metric mAP$_{50}$, so unsurprisingly mAP$_{50}$is maximized for t_{IoU}=0.5. A t_{IoU} of 0.60 leads to a slightly higher mAP, but also a reduced mAP$_{50}$. There is a severe performance drop for t_{IoU}=0.9. This drop goes hand in hand with a sharp drop in the number of TP targets τ which is also likely a part of the reason for the performance drop. The smaller number of TP detections makes it harder to properly fit the the calibration curve and can introduce artifacts from outliers in low density regions.

Impact on calibration curve. In Fig. 1 we plotted the calibration curves for the range of t_{IoU} values and an initial confidence of 0.9. Here we can also observe that t_{IoU}=0.9 breaks the trend of the other thresholds and bends lower for very small IoU values. This is likely an artifact of the Beta calibration function.

Varying t_{IoU} for the calibration metrics. Same as with variation of the t_{IoU} for TP detections of the conditional calibration we can also change the t_{IoU} for the calibration metrics. We show a grid of the resulting calibration metrics in Fig. 2. The Expected Calibration Error (ECE), Adaptive Calibration Error (ACE), and Static Calibration Error (SCE) all follow a similar trend: the respective calibration metric is minimized for if the fitting- and the metric-t_{IoU} are the same. There is, again, a sharp drop-off if one of the t_{IoU} values is 0.9 and the other is not. Otherwise the miscalibration increases with increased distance between the two t_{IoU} values.

t_{IoU}	# τ	mAP↑	mAP$_{50}$↑
0.50	31282	41.36±1.00	**61.28**±1.32
0.60	28068	**41.40**±1.01	61.02±1.33
0.70	26795	41.32±1.02	60.47±1.34
0.80	23855	40.84±1.03	59.11±1.31
0.90	15933	37.70±1.00	53.23±1.53

Table 1. **Comparison of the impact t_{IoU} on the performance of IoU-aware calibration.** We vary the t_{IoU} that used to determine τ—the optimization target for our conditional confidence calibration—from 0.5 to 0.9.

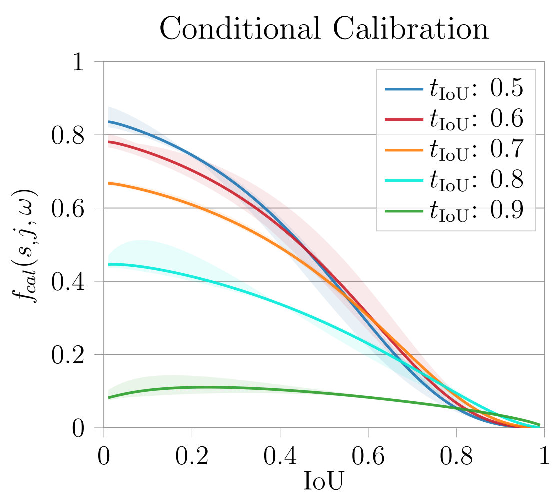
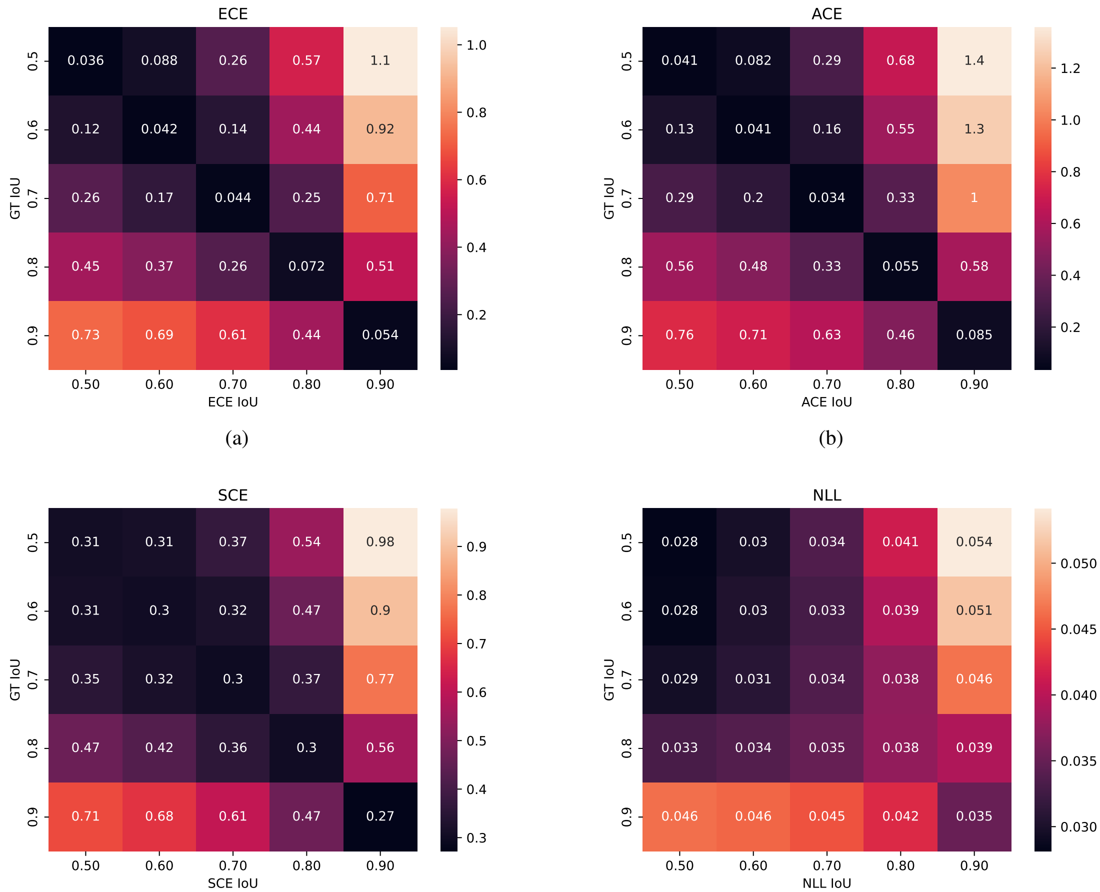


Figure 1. **Comparison of the impact of t_{IoU} for TP detections on the conditional calibration curves.** Shows how confidence of detections is adjusted, depending on the IoU with a more confident detection with initial confidences s=0.9. Confidence intervals in lighter colours.



(a) (b) (c) (d)

Figure 2. **Comparison of t_{IoU} values required for a detection to be considered a TP detection.** On the Y-axis the t_{IoU} for the labels used for fitting the conditional confidence calibration is varied from 0.5 to 0.9, on the X-axis the corresponding t_{IoU} for the labels used for the calibration metric is varied from 0.5 to 0.9. The evaluated calibration metrics are (a) ECE, (b) ACE, (c) SCE, and (d) negative log likelihood (NLL).

Model	Backbone	Settings	Default NMS	Best NMS	Reported mAP	Used implem. mAP
Varifocalnet RN50 [15]	ResNet-50	e:24, DCNv2, FPN	t_{nms} = 0.60	σ= 0.6	44.3	47.8
YOLOX-L [5]	CSP-V5	e:300	t_{nms} = 0.65	t_{nms} =0.7	50.0	49.4
Faster-RCNN RN50 [12]	ResNet-50	e:36, FPN, MS	t_{nms} = 0.70	σ= 0.5	-	40.3
YoloV3-608 [11]	DarkNet-53	e:273	t_{nms} = 0.45	σ= 0.3	33.0	33.7
RetinaNet RN101 [10]	ResNet-50	e:24, MS, FPN	t_{nms} = 0.50	σ=0.6	37.8	38.9
HTC CBNetv2 Swin-L † [8]	Swin-L	e:12, MS	σ = 0.001	σ=0.4	59.1	59.1
EVA Cascade Mask-RCNN † [4]	EVA	e: 24	t_{nms} = 0.60	t_{nms} = 0.50	64.1	63.9
Sparse-RCNN RN50 [14]	ResNet-50	e:36, FPN, MS	none	t_{nms} = 0.80	45.0	45.0
CenterNet HG [16]	Hourglass-104	e:50	none	t_{nms} = 0.80	42.1	40.3
Detr RN50 [2]	ResNet-50	e:150, DCNv2	none	t_{nms} = 0.85	42.0	40.1

Table 2. **Settings for all detectors.** Abbreviations: e refers to the number of training epochs, FPN means the Feature Pyramid Network Neck [9] is used, MS is multi-scale training, and DCN indicates that Deformable Convolutions [3] are used. σ is the hyper-parameter for Gaussian Soft-NMS and t_{nms} the hard threshold for NMS. "Reported mAP" refers to the mAP value reported in the publication that introduced the relevant model. "Used implem." mAP on the other hand refers to the performance of the model implementation we used for our experiments.

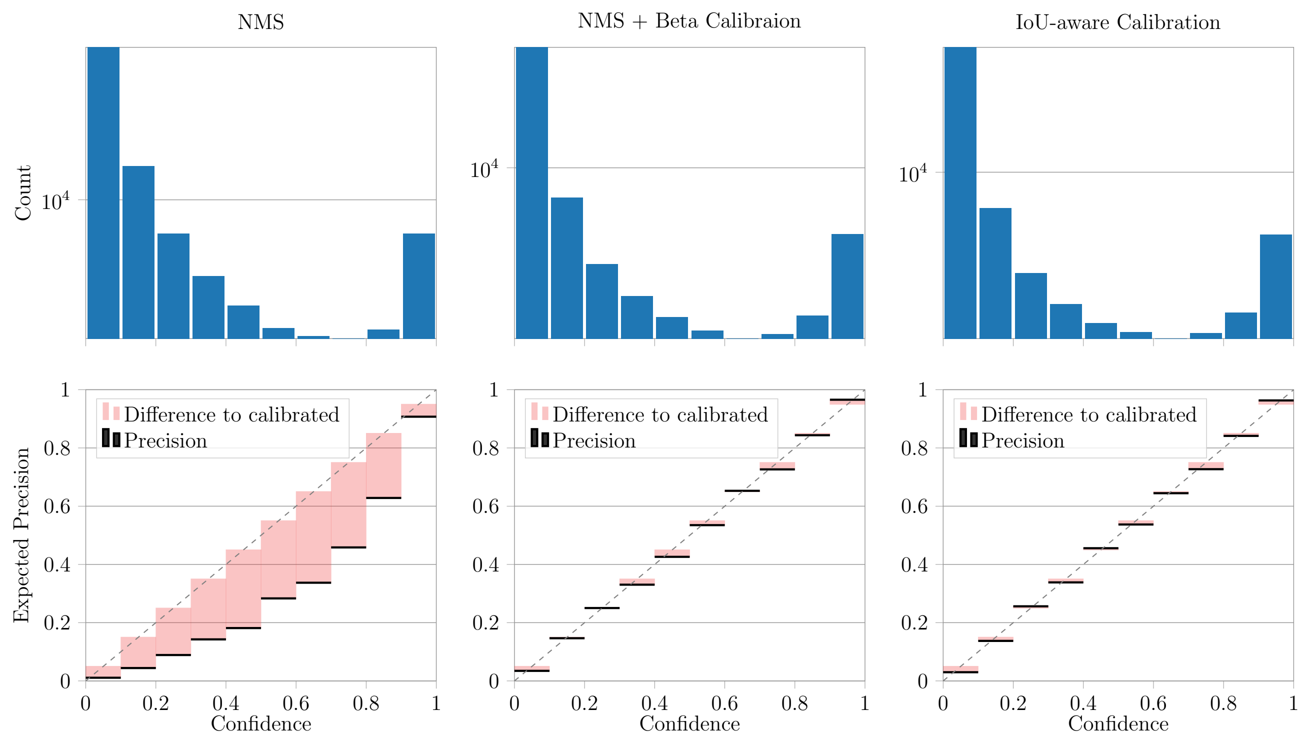


Figure 3. **Reliability diagrams for Faster-RCNN for NMS, NMS with Beta calibration and proposed IoU-aware calibration.** Shows number of detections in each bin on the top and deviation from perfect calibration for each of the 10 bins below.

NMS-Type	parameter	interval start	interval stop	spacing	steps
Non-Maximum Suppression (NMS)	t_{nms}	0.40	0.90	linear	11
Soft-NMS [1]	σ	0.001	0.20	log	20
Weighted Box Fusion [13] (wbf)	t_{nms}	0.50	0.90	linear	11

Table 3. **Settings for NMS hyper-parameter sweep.**

2. Detector Architectures

See Tab. 2 for more detailed settings on the used detection architectures and the best found hyper parameters for NMS. In Tab. 3 we list the intervals for the NMS hyper-parameter sweep for the detectors.

References

[1] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *ICCV*, pages 5561–5569, 2017. 3

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, pages 213–229, 2020. 2

[3] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 764–773, 2017. 2

[4] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva: Exploring the limits of masked visual representation learning at scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19358–19369, 2023. 2

[5] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2

[6] Johannes Gilg, Torben Teepe, Fabian Herzog, and Gerhard Rigoll. The box size confidence bias harms your object detector. In *WACV*, pages 1471–1480, 2023. 1

[7] Fabian Kuppers, Jan Kronenberger, Amirhossein Shantia, and Anselm Haselhoff. Multivariate confidence calibration for object detection. In *CVPRW*, pages 326–327, 2020. 1

[8] Tingting Liang, Xiaojie Chu, Yudong Liu, Yongtao Wang, Zhi Tang, Wei Chu, Jingdong Chen, and Haibin Ling. Cbnet: A composite backbone network architecture for object detection. *TIP*, 31:6893–6906, 2022. 2

[9] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 2

[10] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2980–2988, 2017. 2

[11] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[12] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region

proposal networks. *PAMI*, Jun 2017. 2

[13] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, 2021. 3

[14] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, et al. Sparse r-cnn: End-to-end object detection with learnable proposals. In *CVPR*, pages 14454–14463, 2021. 2

[15] Haoyang Zhang, Ying Wang, Feras Dayoub, and Niko Sunderhauf. Varifocalnet: An iou-aware dense object detector. In *CVPR*, pages 8514–8523, 2021. 2

[16] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 2