Supplementary Material: Technical Appendix

Anonymous submission

Abstract

Supplementary material. The main objective of this document is to present data and figures that further explain the approach. First, using birdviews, we illustrate examples of the criticality score $\kappa(B)$, distance criticality $\kappa_d(B)$, collision distance criticality $\kappa_r(B)$, and collision time criticality $\kappa_t(B)$. Second, we provide additional graphs that describe specific conditions of AP_{crit}, that could not be inserted in the paper becaus of page constraints. Third, we discuss the relations between reliability-weighted precision $P_{\mathcal{R}}$ and safety-weighted recall $R_{\mathcal{S}}$. Fourth, we briefly review the software we developed and released as open source.

1 Bird-view and distance criticality $\kappa(B)$

The following bird-view images explains how our metric model works, in a very practical way, for the computation of $\kappa(B)$. We consider two object detection models models as examples, PGD and SEC, but all models lead to similar conclusions.

The figures below are extracted relying on the nuscene-dev kit 1.1.2, properly modified to visualize our scores. The ego is in the center, at the $(0, 0)$ coordinates. The axes represents distances, in meters. The ground truth (real position and orientation of cars) is in green, the detected cars are instead in blue. Both ground truths and detected vehicles have associated a number, which is either $\kappa(B)$, $\kappa_d(B)$, $\kappa_r(B)$, and $\kappa_t(B)$ depending on what we are investigating. We manually added text labels and red circles to improve readability.

We first consider PGD with $D_{max} = 30$, $R_{max} = 20$, $T_{max} = 8.0$. We select this setting because it is a realistic one. Very intuitively, we believe that it is critical to detect vehicles that are within 30 meters, and/or that are in a colliding trajectories within 20 meters in the next 8 seconds. According to our model, the shortest the distance, the higher the criticality; similarly, the shortest the time to collision, the higher the criticality.

We start from Figure 1. A car is very close to ego, but it is not detected: it has been assigned $\kappa(B) = 0.98$. This car is located at the center of the diagram, and it is circled in red. Another one is very close, but in "a bit less dangerous" situation: $\kappa(B) = 0.89$. A third one is within $D_{max} = 30$ meters, but headed in a different direction, so it gets a mild criticality score $\kappa(B) = 0.60$. Instead, there are other less critical missed detections in the upper and lower parts of the image. These are farther than $D_{max} = 30$ meters, and headed in non-colliding trajectories: these are irrelevant, so they are worth $\kappa_d(B) = 0.00$.

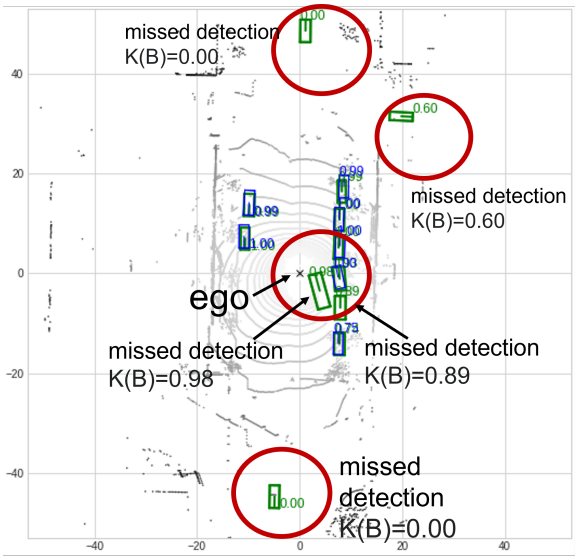


Figure 1: $\kappa(B) = 0.98$ and $\kappa(B) = 0.89$ for two dangerous missed detections from PGD. *Best viewed in color.*

In Figure 2, the same scene is analyzed using SEC (we just extract a part of the birdview to improve readability). This time, the most dangerous missed detection of Figure 1 is properly detected. Instead, there are false positives (see the many blue vehicles that are not overlapping to a green vehicle). Some of them may impact the driving task, e.g., they may require some steering of ego or in any case they should be taken into consideration. Others are irrelevant.

Next, we explore the contribution of the distance criticality $\kappa_d(B)$, the collision distance criticality $\kappa_r(B)$, and the collision time criticality $\kappa_t(B)$ to the criticality score $\kappa(B)$. We consider SEC with $D_{max} = 15$, $R_{max} = 20$, $T_{max} = 10$ in Figure 3, where we show $\kappa(B)$ as usual.

Figure 4 shows the values of $\kappa_d(B) = 0$ with $D = 15$ meters from ego. Cars farther than 15 meters from ego are assigned $\kappa_d(B) = 0$; the closest to ego, the higher the values.

Figure 5 shows the values of $\kappa_r(B)$ with $R_{max} = 20$

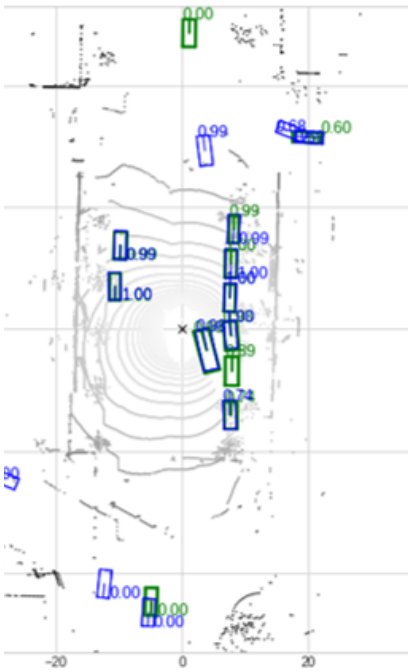


Figure 2: $\kappa(B)$ rates false positives from SECFPN. Clearly there are some false positives (blue vehicles without any green vehicle overlapping), but only two of them (with assigned $\kappa(B) \geq 0$ may interfere with ego. *Best viewed in color.*

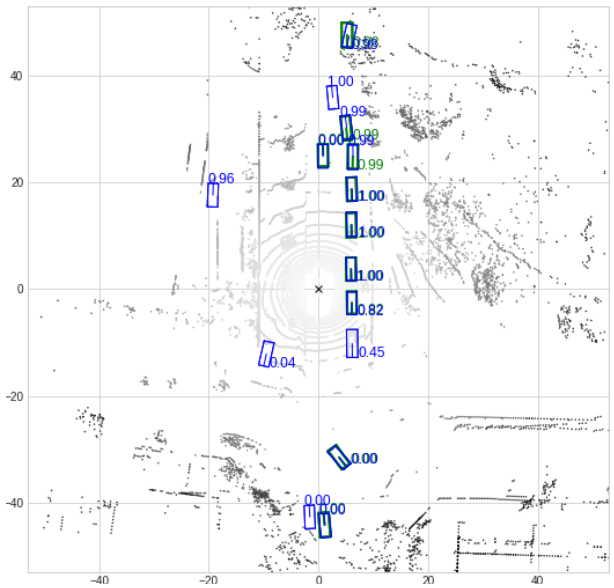


Figure 3: $\kappa(B)$ computed for SEC with $D_{max} = 15$, $R_{max} = 20$, $T_{max} = 10$. *Best viewed in color.*

meters from ego. Cars with a trajectory passing closer to ego than R_{max} are assigned $\kappa_r(B) > 0$; the closest to ego, the higher the values, meaning that those passing very close are at risk of collision. Note that objects can be assigned a high

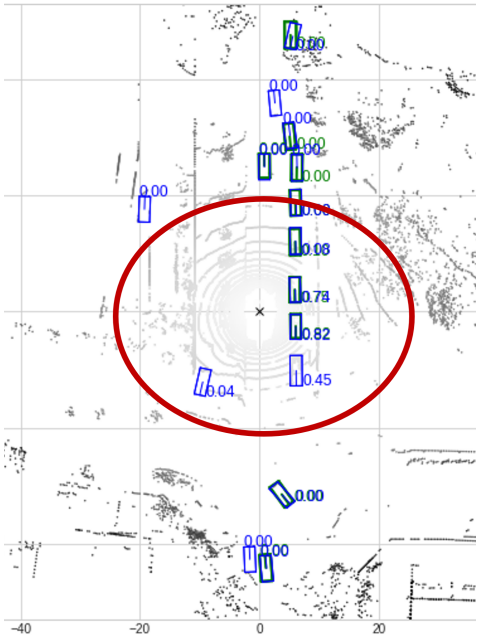


Figure 4: $\kappa_d(B)$ computed for SEC with $D_{max} = 15$, $R_{max} = 20$, $T_{max} = 10$. The red circle is an area of approximately $D_{max} = 15$ meters around ego. Values farther than this are assigned $\kappa_d(B) = 0$. *Best viewed in color.*

$\kappa_r(B)$ value even if they do to appear to actually collide with ego. It is the case of the line of cars in the upper right part of 5: even if they are proceeding straight, they are passing very close on the right of ego, and thus the indicator of potential collision is close to 1.0.

Similarly, Figure 6 shows the values of $\kappa_t(B)$ with $T = 10$ seconds. Cars which may enter in a collision within 10 seconds are assigned $\kappa_t(B) > 0$.

2 A bit more on AP_{crit} trends

We further study AP_{crit} for different values of D_{max}, R_{max}, T_{max}. We pick the object detector REG1.6 with $l = 2.0$ (the others have analogous trends). In Figure 7 we show the AP_{crit} when $R_{max} = 20$; the figure clearly shows how the highest AP_{crit} values are achieved when D_{max} is set in the range $[20, 30]$. This is possibly due to the fact that setting D_{max} very low excludes a lot of "easy" (i.e., close) objects from the relevant ones, thus deteriorating AP_{crit}. Conversely, when D_{max} becomes much greater than R_{max}, a lot of distant but not relevant objects are included, which are unlikely to reach a collision point closer than R_{max}. Note that $AP = 0.89$, which is represented as a flat grey surface in the figure. Figure 7 shows that AP_{crit} is in general higher than AP. This is expected, because the AP_{crit} gives less weight to many vehicles that are harder to detect, e.g., those at a farther distance from ego.

To provide a summarizing view, in Figure 8 we show a 4D plot of SSNREG with $l = 1.0$: configurations $(D_{max}, R_{max}, T_{max})$ producing higher values of AP_{crit} are represented with larger and darker dots. Figure 8 confirms

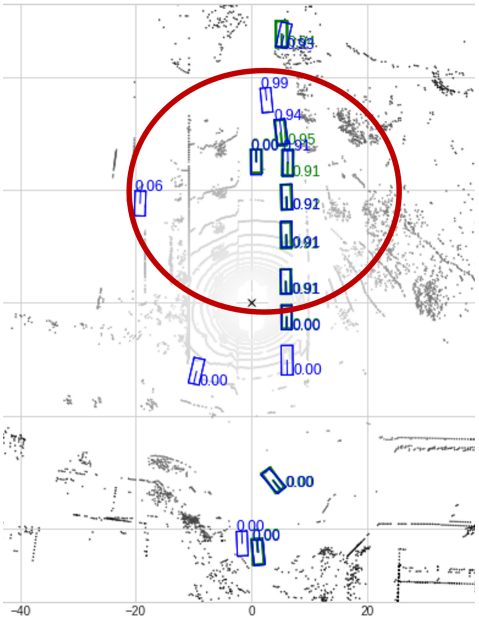


Figure 5: $\kappa_r(B)$ computed for SEC with $D_{max} = 15$, $R_{max} = 20$, $T_{max} = 10$. The red circle is an area of approximately $R_{max} = 20$ meters from ego. Objects whose trajectories may overlap with ego trajectories have assigned $\kappa_r(B) \geq 0$. Note that computation of $\kappa_r(B)$ requires the direction of the vehicles estimated from the velocity vector, which is not plotted. *Best viewed in color.*

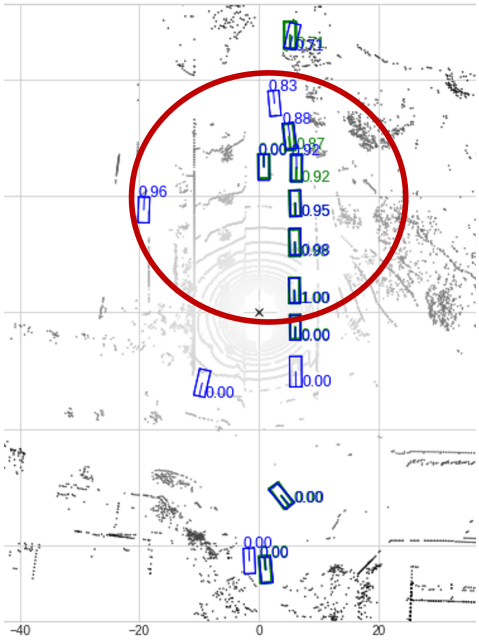


Figure 6: $\kappa_t(B)$ computed for SEC with $D = 15$, $I = 20$, $T = 10$. Objects which may collide with ego within 10 seconds have assigned $\kappa_t(B) \geq 0$. Note that computation of $\kappa_t(B)$ requires the velocity, which is not plotted. *Best viewed in color.*

the previous consideration on the evolution of AP_{crit} values: the highest AP_{crit} values are obtained with low values of R_{max} and T_{max}, and with D_{max} in the interval $[20, 30]$. A similar trend is observed for the other object detectors and l values.

3 Detailed results on $P_{\mathcal{R}}$ and $R_{\mathcal{S}}$

To discuss the relations between reliability-weighted precision $P_{\mathcal{R}}$ and safety-weighted recall $R_{\mathcal{S}}$, we use the same approach of the traditional precision-recall curve. In Figure 9, we plot the nine object detectors for $(D_{max}, R_{max}, T_{max}) = (20, 15, 8)$ with $l = 0.5$ (left) and $l = 2$ (right). We select such triple as a representative configuration because, intuitively, it is one of the settings that may have practical use: it includes objects that are in the close surroundings of ego while it is roving around the city. We also plot the precision-recall curve for REG1.6, which is the best object detector when traditional P and R are considered. The trends are similar for other triples $(D_{max}, R_{max}, T_{max})$ and l values, and match the expected behavior of a precision-recall curve. In agreement with (Caesar et al. 2020), cases in which recall or precision is less than 0.1 are removed; this avoids showing the noise commonly seen in low precision and low recall regions.

We notice that the higher the l, the higher the value of $P_{\mathcal{R}}$ given a target $R_{\mathcal{S}}$. This can be observed comparing the difference between Figure 9a and Figure 9b, and it is clearly visible for FCOS and PGD. This is expected, because with

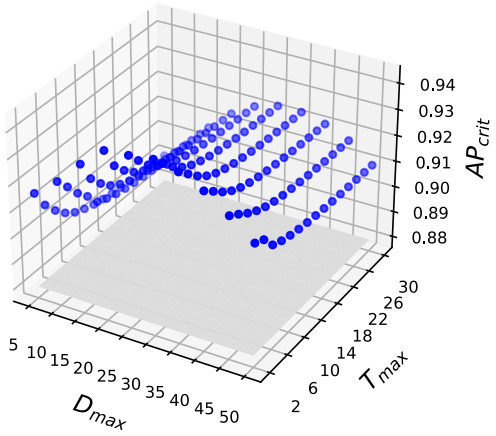


Figure 7: AP_{crit} measured on REG1.6 with $l = 4.0$ and $R_{max} = 20$, for the different D_{max} and T_{max}. *Best viewed in color.*

a higher l value, the number of bounding boxes that are considered correct detections is increased.

In the paper we presented a subset of results on the relations between $P_{\mathcal{R}}$ and $R_{\mathcal{S}}$ for high values of $R_{\mathcal{S}}$ (safety-weighted recall), which are of particular interest in the reference domain of this work. The results for some l were removed from the paper for space constraints, but we believe they are helpful to investigate the $P_{\mathcal{R}}$ (reliability-weighted

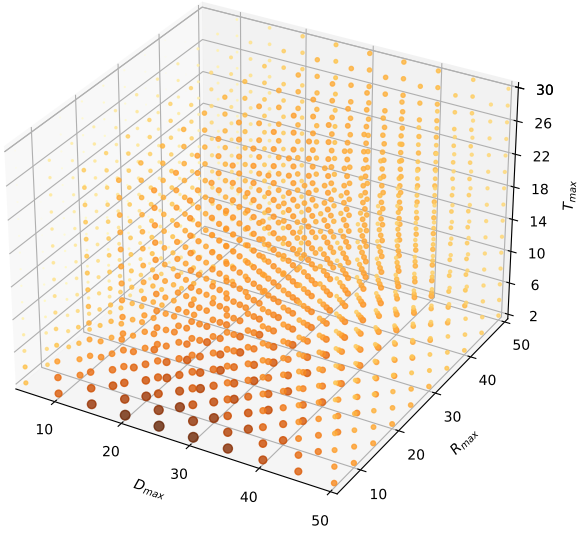
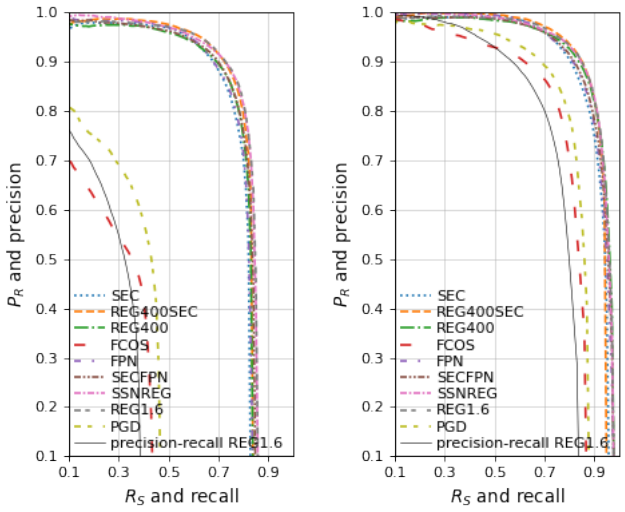


Figure 8: AP_{crit} for SSNREG with $l = 1.0$ and different configurations of $(D_{max}, R_{max}, T_{max})$. The darker and larger the dots, the higher the AP_{crit}. *(best viewed in color)*



(a) $l = 0.5$ (b) $l = 2$

Figure 9: Relations between $P_{\mathcal{R}}$ and $R_{\mathcal{S}}$ for all object detectors with $l \in \{0.5, 2\}$ and configuration $(20, 15, 8)$. Precision-recall curve is drawn for REG1.6. *Best viewed in color.*

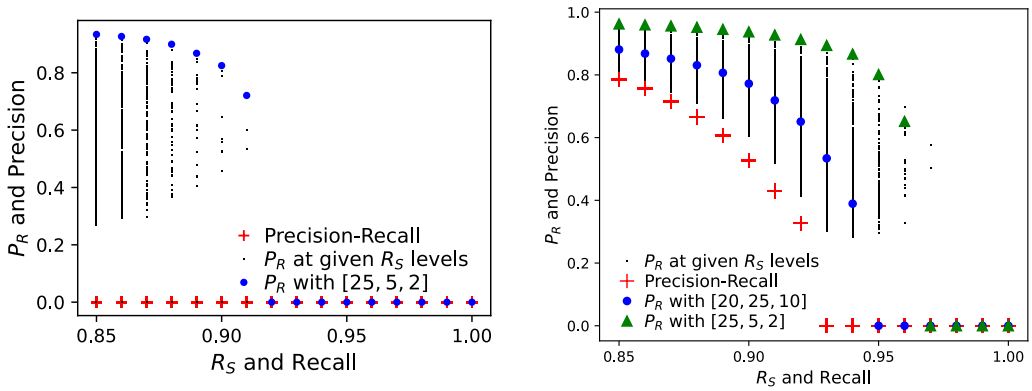
precision) that we achieve when safety is enforced thanks to a high $R_{\mathcal{S}}$. This corresponds to answering the question "given a safety target on the detection, what is the possibility of driving the car with good mission reliability, i.e., without being forced to interrupt the driving continuously because of false positives?". Of course, the safest condition would be $R_{\mathcal{S}} = 1$, but $P_{\mathcal{R}}$ is typically 0 in such cases; still, a very high $R_{\mathcal{S}}$ is necessary to enforce safety of the detection.

For traditional metrics, when the recall R increases, the precision P quickly drops to 0, and it is always zero in the case of Figure 10a where $l = 0.5$. This means that REG1.6 can offer a high recall, i.e., high ability of detecting all the objects, only at the cost of many false positives: this is clearly of little or no use in practice. Instead, if we restrict the scope of the object detector thanks to our model, we reach different conclusions. For example, consider the case $l = 0.5$ (Figure 10a). Even with $R_{\mathcal{S}} \geq 0.9$, there are some configurations in which $P_{\mathcal{R}} > 0.8$, which is clearly a much more comforting result, showing confidence in the detection at least to some extent. On the other hand, the best performing triple, represented with the blue dots in Figure 10a, may be not practical, because it is computed applying small spatial and temporal distances of the objects from ego. The same reasoning applies to the other figures, with small differences: for example, in Figure 10c and Figure 10d the best performing configuration, for the highest values of $R_{\mathcal{S}}$, is $(5, 20, 2)$ and it is represented by green triangles. Still, in the figures, the numerous black dots can be easily spotted: these represents the triples $(D_{max}, R_{max}, T_{max})$ that achieve $P_{\mathcal{R}} \geq 0$ for $R_{\mathcal{S}}$ values above 0.85, and in many cases also when $R_{\mathcal{S}} \geq 0.90$. For example, Figure 9 is an example of configuration $(20, 15, 8)$ which is amongst the black dots just mentioned.

4 Software to reproduce results

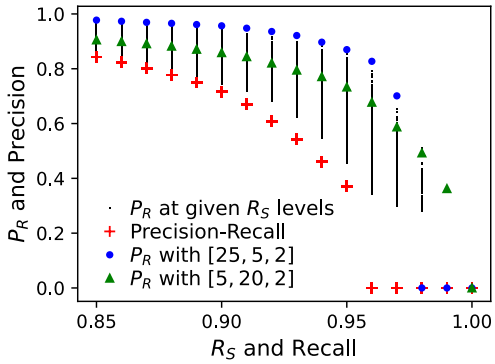
The software submitted is from a github repository we created, not mentioned to maintain anonymity, and released open source. The software, uploaded as supplementary material, contains the following:

- The README file contains detailed instructions to execute the software and reproduce the same exact results of the paper.
- A folder, named *eval*, contains the modified nuscenes-dev library. The modifications allow computing the AP_{crit} and the related metrics.
- three jupyter notebooks, to be executed in this order.
  - The first one is MMDetection3D.ipynb. This simply runs the detectors of mmdetection3D, and collects results in a format which is compatible for nuscenes-dev. Note that we used detectors from mmdetection3D, but *any* detector is fine, as long as it produces outputs compatible with the nuScenes format.
  - Then, compute_APCRIT.ipynb allows computing AP_{crit} and all the metrics that have been presented in the paper. It computes them for multiple cases of D_{max}, I_{max}, T_{max}. You can choose the triples of D_{max}, I_{max}, T_{max} you prefer. Default values are the same as in the paper.
  - Last, Analysis_of_data.ipynb. This is just data analysis. It explores the values computed by compute_APCRIT.ipynb, to get insights, extract tables, extract figures. For example, the results and graphs presented in the paper are extracted from this jupyter notebook.

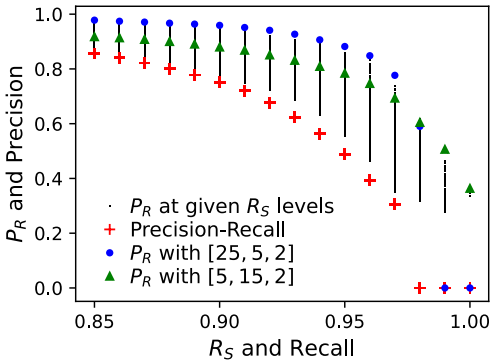


(a) $l = 0.5$

(b) $l = 1.0$



(c) $l = 2.0$



(d) $l = 4.0$

Figure 10: $P_{\mathcal{R}}$, $R_{\mathcal{S}}$, P and R for REG1.6 when $R_{\mathcal{S}} \geq 0.85$ and $R \geq 0.85$.

References

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.