ALPI: Auto-Labeller with Proxy Injection for 3D Object Detection using 2D Labels Only
-Supplementary Material-

Saad Lahlali Nicolas Granger Hervé Le Borgne Quoc-Cuong Pham

Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France

firstname.lastname@cea.fr

1. Size priors

Implementation details In our approach, size priors are utilized for various tasks, including object depth estimation, background extraction, and proxy object generation. These priors were obtained by searching the internet using class names, and the resulting statistical data is consistently applied across all datasets, as detailed in Table 1. For certain classes, such as *Car* and *Pedestrian*, relevant subcategories were provided. When generating proxy objects, we take into account the dimension's distribution of each subcategory. For example, the *Car* class includes subcategories like *City Car* and *Passenger Van*, which differ in size. This allows us to create proxy objects with more realistic size correlations; for instance, longer cars tend to be wider and taller as well. Since the distribution of these subcategories is unknown, we uniformly sample a subcategory when generating proxy objects for the classes *Car* and *Pedestrian*. We then apply a Gaussian distribution based on the mean and standard deviation of that subcategory's dimensions. For classes without identified subcategories, we use the overall mean and standard deviation for the entire class. It is important to note that this subcategory distinction is used exclusively for proxy object generation for the classes *Car* and *Pedestrian*. In all other parts of our method, we use the overall distribution of the class, applying the mean and standard deviation across the entire class when the subcategory is unknown.

Size priors from the validation set. An alternative to extracting the size priors statistics from the internet would have been to get them from the validation set. In Table 2, we present the results of this study which shows that using the statistics from the validation set lower the performances for the class *Car* on KITTI since the diversity of the subcategories offer better realism between dimensions when generating proxy objects.

2. Quality of the object's depth estimation

In Section 3.1, we use a simple method to examine the frustum point cloud and estimate a 3D center in which insert a proxy object after removing points belonging to the original object. In order to prove that this simple approximation is acceptable for the purpuse of building a realistic background before injecting the proxy object we present in Table 3 an experiment where

3. Depth-normalized property of our proposed 2D loss

In Section 3.2, we examined the scenario where a predicted Box_{3D} is shifted from its ground truth by a predetermined distance and translate both boxes across different depth positions. The resulting 3D Intersection over Union (IoU) error remains constant, as shown in Figure 1a. Conversely, the amplitude of the L_{2D} loss varies depending on the object's depth in the scene due to 2D projection effects. Additionally, we observed that when the 2D detection loss lacks depth normalization, it leads to poor alignment with the 3D detection task. We define the alignment as $1 - |Err^{2D} - IoU_{3D}|$, utilizing the 2D losses to measure the 2D error Err^{2D}. In Figure 1b, it is evident that our depth-normalized detection loss achieves better alignment compared to the conventional 2D loss.

4. Effect of pseudo-labeling iteration on all classes

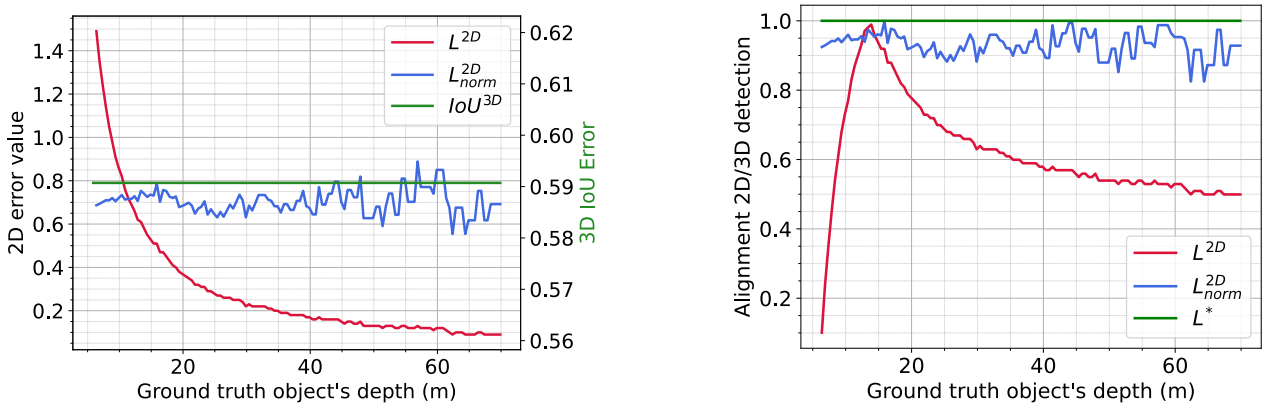
In Table 4, we present the performance of our method on the KITTI validation set for all classes following each pseudo-labeling iteration. The results demonstrate that ALPI gradually improves the quality of generated pseudo-labels. Visualizations are provided in Figure 2 and Figure 3.

Statistic	Car [1]			City car [1]			Small car [1]			Compact car [1]			Familly car [1]			Executive car [1]			Luxury car [1]			Sprorts car [1]		
	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.
Mean	4.35	1.56	1.84	3.07	1.51	1.76	3.07	1.51	1.76	4.22	1.5	1.8	4.19	1.45	2.1	4.79	1.44	1.85	4.99	1.42	1.92	4.31	1.28	1.85
Std	0.55	0.12	0.08	0.72	0.03	0.63	0.11	0.04	0.03	0.08	0.04	0.02	0.85	0.04	0.87	0.05	0.03	0.02	0.07	0.03	0.02	0.22	0.08	0.08

Statistic	MPV [1]			Small SUV [1]			Compact SUV [1]			Mid-size SUV [1]			Large SUV [1]			Pick-up [1]			Passenger Van [1]			Estate car [1]		
	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.
Mean	4.63	1.48	1.81	4.7	1.76	1.87	3.93	1.53	1.71	4.23	1.56	1.8	4.54	1.63	1.83	4.73	1.68	1.91	5.35	1.83	1.86	4.43	1.84	1.84
Std	0.04	0.03	0.02	0.35	0.17	0.07	0.16	0.04	0.06	0.03	0.04	0.01	0.03	0.03	0.02	0.02	0.09	0.04	0.12	0.04	0.03	0.05	0.03	0.02

Statistic	Pedestrian [7]			Pedestrian adult [7]			Pedestrian kid [7]			Cyclist/Bike/Motor [6]			Bus [4]			Trailer [5]			T.C. [2]			Barrier [3]		
	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.	L.	W.	H.
Mean	0.70	0.70	1.62	0.70	0.70	1.73	0.70	0.70	1.52	1.8	0.64	1.73	15.75	3.6	2.5	6	2	2	0.7	0.5	0.5	1.5	0.6	0.4
Std	0.2	0.2	0.06	0.2	0.2	0.09	0.2	0.2	0.05	0.2	0.2	0.09	0.2	0.2	0.2	1.5	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2

Table 1. Mean and standard deviation of size dimensions (height, width, length) for all classes according to their class name in the datasets KITTI and nuScenes. The two upper tables are used for the class *Car*. *Pedestrian adult* and *Pedestrian kid* are used for the *class Pedestrian*. When diverse data is not available to compute a standard deviation, we used a fill value of 0.2.



(a) Loss values

(b) Alignment between 2D and 3D detection

Figure 1. When employing the usual smooth L1 (L_{2D}), a constant 3D IoU error across depth does not translate to a constant 2D loss which causes a poor alignment between 2D/3D detection across different depths. However, with our L^{2D}_{norm}, the depth-independence property is better respected, bringing the alignment closer to the unavailable perfect loss L^{*}.

Size prior source	easy	mod.	hard
Val set	90.01	78.78	73.34
Internet	90.48	79.43	75.81

Table 2. 3D object detection mAP (%) on KITTI **val** set, with size priors extracted from the validation set and the internet.

Background extraction method	easy	mod.	hard
Ground truth	90.59	79.64	75.83
Our estimation	90.48	79.43	75.81

Table 3. 3D object detection mAP (%) on KITTI **val** set, with size priors extracted from the validation set and the internet.

References

[1] Automobile Dimension. Car dimensions, 2024. Accessed on September 1st, 2024. 2

[2] Barriersdirect. Traffic Cone Guide and Sizing Standards, 2023. Accessed on September 1st, 2024. 2

[3] Deeproot. Root Barriers: Sizes and Types, 2024. Accessed on September 1st, 2024. 2

[4] Group Transport Australia. How long is a bus?, 2024. Accessed on September 1st, 2024. 2

[5] Joel Steele. Trailer Sizes Guide, 2024. Accessed on September 1st, 2024. 2

[6] The best bike lock. Trailer Sizes Guide, 2022. Accessed on September 1st, 2024. 2

[7] World Population Review. Average Height by Country 2024, 2024. Accessed on September 1st, 2024. 2

Successive steps		**Car** (IoU=0.7)			**Ped.** (IoU=0.5)			**Cyclist** (IoU=0.5)		
		Easy	Mod.	Hard	Easy	Mod.	Hard	Easy	Mod.	Hard
ALPI	Iteration 0	68.94	64.8	56.73	54.58	49.31	45.83	51.94	45.16	45.55
	Iteration 1	87.10	76.49	74.67	56.65	50.78	50.91	57.24	56.15	51.01
	Iteration 2	87.84	77.12	76.48	56.34	51.3	50.37	68.43	60.09	60.98
	Refinement	88.97	77.80	77.48	56.41	51.6	51.12	70.49	61.37	60.91

Table 4. 3D object detection AP (%) on KITTI **val** set.

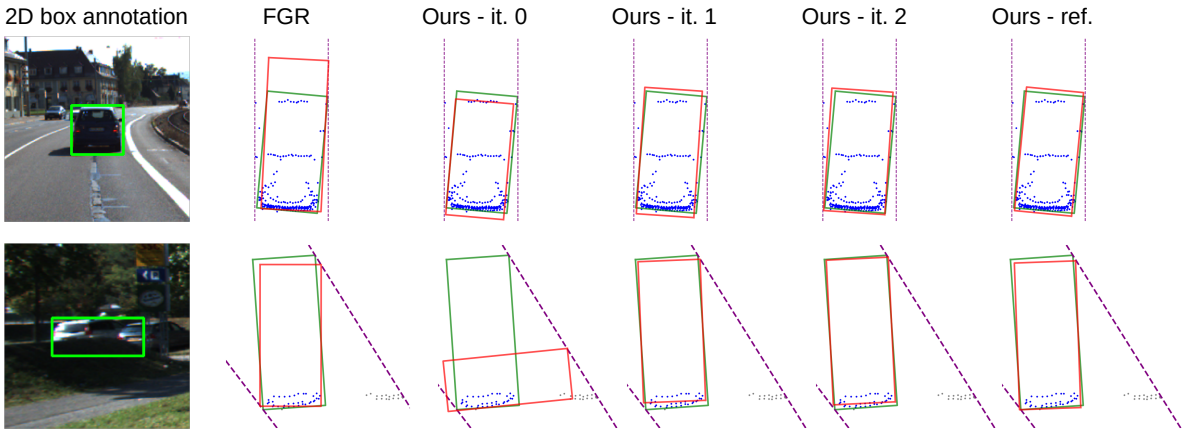


Figure 2. Comparison of annotations produced by our method and FGR. Predicted 3D boxes are drawn in red while ground truth boxes are in green. The 2D box annotations used during training are shown in the left. For better visualisation, we color in blue the points belonging to the object of interest, in grey occluding object and in purple the frustum lines drawn using 2D box annotations.

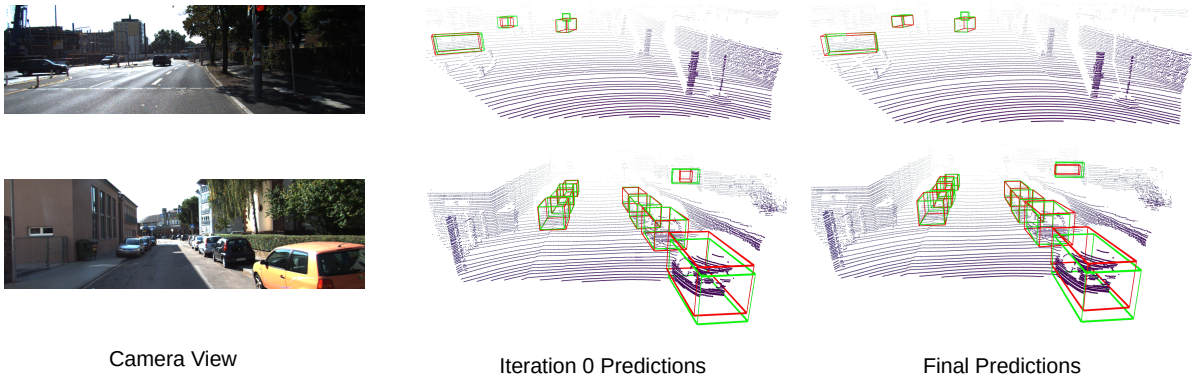


Figure 3. Sample predictions of our method. Predicted bounding boxes are drawn in red while ground truth boxes are in green.