Multi-Modal Annotation of Regulatory Road Signs along Navigation Paths for Autonomous Driving

Alessandro De Luca*, Ottavia Belotti*, Gabriele Caslini†, Riccardo Pieroni*, Matteo Corno* and Sergio M. Savaresi*
*†Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano, Milan, Italy
Email: {name}.{surname}@*{polimi}/†{mail.polimi}.it

***Abstract*—This work presents a fully automated offline framework to populate navigation paths (i.e. reference trajectories) with traffic rules information in the context of Autonomous Driving. Rather than reconstructing full scene geometry, our pipeline focuses on the functional association of traffic regulations (e.g., stop signs, speed limits, crosswalks) to the specific path nodes where the vehicle must react. We propose a multi-modal approach that fuses high-recall 2D object detection with LiDAR-based depth projection to recover the 3D position of regulatory elements. Crucially, we introduce a non-causal temporal aggregation strategy that utilizes the full history of a traversal to filter false positives and refine the logical placement of attributes. Experimental results on a proprietary dataset of Italian urban roads demonstrate that our method achieves high annotation accuracy and geometric consistency, offering a scalable solution to automate the semantic enrichment of autonomous navigation graphs.**

***Index Terms*—Traffic signs, Road markings, Autonomous driving, Navigation path annotation, Vehicle mapping**

I. Introduction

Autonomous navigation in complex urban environments requires a robust understanding of traffic regulations, such as speed limits, right-of-way rules, and stop lines. To manage this complexity, autonomous systems can rely on augmented navigation paths (i.e, reference trajectories enriched with semantic attributes that explicitly dictate vehicle behavior at specific locations), as in [1], [2]. While this path-based formalism simplifies online planning by embedding rules directly into the navigation graph, generating these annotations remains a significant bottleneck. Currently, this process relies heavily on manual labeling, which is costly, unscalable, and prone to error. This work presents a fully automated offline pipeline to generate these semantic annotations without human intervention. Given a raw sensor recording and a reference path, our system detects, localizes, and logically associates traffic regulations to the navigation nodes where the ego-vehicle must react. Unlike geometric HD mapping approaches that aim to reconstruct the full physical scene, our method focuses on functional annotation, conveying information on *how* the vehicle should act along the path rather than *where* exactly the traffic signs are placed. We propose a multi-modal architecture that combines the high recall of 2D deep learning detectors with the spatial precision of LiDAR point clouds. By operating offline, we leverage non-causal temporal aggregation, by utilizing the entire history of a traversal to filter transient false positives and refining the 3D position of regulatory elements. This ensures that the final annotations on the path are robust, geometrically consistent, and ready for deployment, while reducing the need for human intervention in the map updating process. The main contributions of this paper are:

- We propose a modular, multi-modal pipeline for automatic ego-path annotation that leverages camera, LiDAR, and ego-pose data to robustly localize and classify traffic-related infrastructure.
- We introduce a hierarchical annotation strategy that aggregates redundant detections across time and modalities, explicitly accounting for the spatial placement conventions of traffic signs (TS) in Italian urban environments.
- We propose a safety-aware evaluation methodology based on asymmetric distance validity windows and behavior-consistent analysis. The approach assesses both the accuracy of attribute placement and the robustness to localization errors, with particular emphasis on preserving correct driving behavior under misclassification. The pipeline is validated on 14 km of manually annotated Italian urban roads.

II. Related Works

A. Paradigms in Autonomous Navigation

Autonomous navigation systems fundamentally rely on a representation of the environment to plan safe trajectories. Broadly, the literature presents two dominant paradigms for utilizing this prior knowledge: *Geometric HD Mapping* and *Reference Path Following*.

a) Geometric HD Mapping: The standard approach relies on dense High-Definition (HD) maps that explicitly reconstruct the physical geometry of the static environment, including lane boundaries, curbs, and traffic signs [3]. In this paradigm, the global planner utilizes the map primarily for localization, while the local planner must dynamically compute a trajectory that adheres to traffic rules by spatially reasoning about the vehicle's position relative to map features.

b) Reference Path Following: Alternatively, [1], [2] shift the complexity from online reasoning to offline precomputation by utilizing *augmented navigation paths*. In this paradigm, the "global plan" is not just a route, but a reference

trajectory enriched with semantic attributes. While traditional HD mapping focuses on geometric scene reconstruction, autonomous planning can also rely on these navigation paths (i.e., reference trajectories that guide the vehicle's high-level intent). [2] demonstrated that explicit navigation paths are critical for conditioning planners, serving as a strong prior for generating local target points. However, these paths are only effective if they are semantically rich (i.e., encoding where to stop or yield).

Distinct from geometric approaches, where the vehicle must deduce behavior from the environment's shape, this paradigm requires the path itself to carry functional attributes. Consequently, the challenge shifts from *reconstructing* the world to *annotating* the path nodes where specific regulations apply.

B. Automated Semantic Annotation

Populating these navigation paths with semantic attributes is traditionally a labor-intensive manual process. To automate this, the literature has diverged into online and offline strategies.

a) Online Vectorization: Current online mapping methods, such as MapTR [4] or StreamMapNet [5], attempt to generate this topology and its attributes in real-time from onboard sensors. While promising for scalability, these methods suffer from occlusion and causality constraints, as they cannot correct a noisy detection at time t using information available at time $t+n$. This temporal instability makes them less suitable for generating the ground truth quality required for a static reference path.

b) Offline Aggregation: Offline pipelines, conversely, allow for non-causal processing. [6] recently proposed an offline framework for traffic sign annotation using Structure-from-Motion (SfM) to aggregate detections over time. However, their approach focuses on generating 3D bounding boxes for a geometric map. Our work aligns with the offline philosophy but targets the functional enrichment of the navigation path. By utilizing non-causal temporal aggregation, we associate regulations to the specific path nodes where the vehicle must react, ensuring the reference path contains the robust behavioral attributes required for the safety-critical planning described in [1].

C. Multi-Modal Perception for Localization

Reliably associating a traffic regulation with a specific path node requires precise 3D localization of the road asset. While visual detectors like YOLO [7] provide high-recall 2D recognition, recovering accurate depth is challenging. Purely vision-based approaches, such as IPM [8], fail on non-planar vertical objects, while SfM-based methods [6] struggle in textureless environments or varying lighting conditions. To ensure robustness, we adopt a sensor-fusion strategy. By projecting accumulated LiDAR point clouds into 2D frustums generated by the visual detector, we recover precise metric depth even for small objects like traffic signs. This allows us to spatially lock the regulation to the correct segment of the ego-path with a precision that monocular methods cannot guarantee.

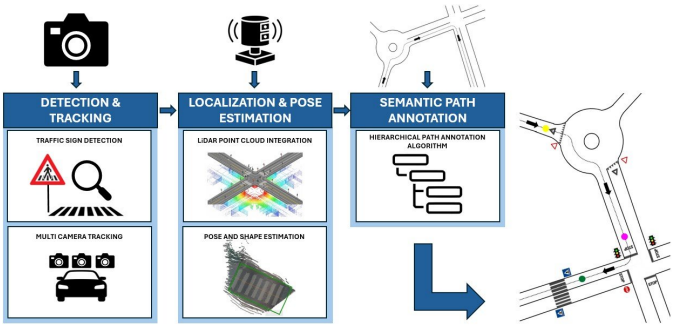


Fig. 1: Complete automatic annotation pipeline

III. Methods

The objective of the proposed pipeline is to automatically generate a semantic path for a given route using only sensor data acquired during a manually driven traversal. The system is composed of three sequential modules (Fig. 1). First, a detection and tracking module identifies instances of both vertical and horizontal traffic signs along the route. Second, a filtering module exploits vehicle ego-pose information, traffic sign detections, and LiDAR point clouds to determine which signs are relevant to the vehicle's driven path and to estimate their position. Finally, an annotation module applies a hierarchical reasoning strategy, based on the rules defined in the Italian Road Code [9], to aggregate redundant information collected over time and assign the correct semantic label such as a stop, yield, crosswalk, roundabout, or traffic light, thereby encoding the expected driving behavior at each location. In the context of this work, the path is represented as a sequence of nodes obtained by discretizing the vehicle trajectory at a spatial resolution of 0.5 m, as described in [1].

A. Detection and Tracking

The first stage of the pipeline performs detection and tracking of both vertical and horizontal traffic signs present in the recording. A YOLOv11-l detector [7] is adopted to jointly predict all traffic sign classes within a unified detection head. Training data consist of a combination of publicly available European benchmarks (DFG [10], MTSD [11], BTSD [12], STSD [13], RTSD [14], and PTSD [15], all unified under a common annotation scheme and further enriched to account for missing signs of interest) and proprietary recordings. The resulting dataset comprises 20,930 frames and more than 34,000 annotations across 31 traffic sign classes. To exploit temporal consistency and the three-camera setup, a tracking-by-detection strategy is employed. Detections from all cameras are jointly processed by a single BoT-SORT-based tracker [16], which leverages global motion compensation and re-identification to handle camera motion and temporary occlusions. While multi-view tracking improves recall, it can introduce false positives from lateral intersections, where signs unrelated to the ego vehicle are detected. To mitigate this effect, a three-camera selection layer filters out detections that are not relevant to the ego lane. Specifically, the tracking layer

retains all objects detected at least once in the central camera. Tracklets originating from lateral cameras are preserved only if they are spatially close to a central-camera track, as measured by a class-dependent centroid distance threshold, and if they are semantically compatible, either by sharing the same class as the central object or by belonging to a predefined set of informative signs.

B. Localization and Pose Estimation

To estimate traffic sign positions along the path, the pipeline adopts a multi-stage localization and pose estimation procedure. Sensor data and ego-pose information are synchronized using the LiDAR clock as reference, associating each LiDAR frame with the temporally closest camera image and ego pose within a fixed tolerance of 0.02 s. This produces synchronized sensor triplets used for 3D reconstruction. For each detected traffic sign, the corresponding image bounding box is projected into the LiDAR frame using the camera intrinsics $\mathbf{K}$ and extrinsics $\mathbf{H}$, defining a 3D frustum. LiDAR points within this frustum are extracted to estimate the sign's 3D displacement, following the approach of [17]. Over time, these measurements are aggregated in global coordinates using ego-pose information, yielding a consolidated point cloud for each vertical and horizontal traffic sign (Fig. 2).

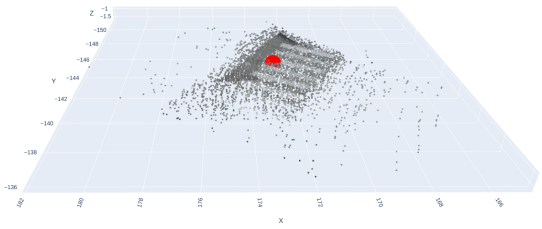


Fig. 2: Aggregated point cloud from tracked frames for a crosswalk sample, shown in ENU coordinates (meters). The red marker indicates the mean point, and each point in the cloud is colored using the RGB values of its corresponding image pixel.

Category-specific geometric filtering is then applied to refine the position estimates. For **vertical traffic signs**, a two-stage k-means clustering ($k = 2$) progressively removes outliers, followed by size and distance filtering (minimum bounding box of 400×400 pixels, maximum distance of 25 m). The point cloud is height-centered to mitigate residual vertical errors, and RANSAC is used to estimate the sign plane and its normal direction $\hat{n}$. Signs oriented away from the vehicle are discarded by evaluating the angle between $\hat{n}$ and the vector connecting the sign center to the vehicle pose at first detection $\hat{d}_c$, in particular it is discarded if $||\hat{n} \times \hat{d}_c|| < \sin(30°)$. For **horizontal traffic signs**, points beyond 25 m or with insufficient LiDAR support are removed. Height-based clustering is applied when excessive vertical dispersion is detected, followed by height centering. To correct trapezoidal distortions introduced by camera–LiDAR projection, points are transformed into a local coordinate frame aligned with the ego-path tangent and normal. Symmetric quantile trimming is then applied to recover a rectangular shape (Fig. 3), specifically by setting $\alpha_t = 0.06$ and $\alpha_n = 0.02$, points lying in the extreme tails of the spatial distribution are discarded, resulting in the set $I = \{i \mid Q_{\alpha/2} \leq \{u_i, v_i\} \leq Q_{1-\alpha/2}\}$. Then RANSAC is used to estimate the supporting road plane while accounting for road slope.

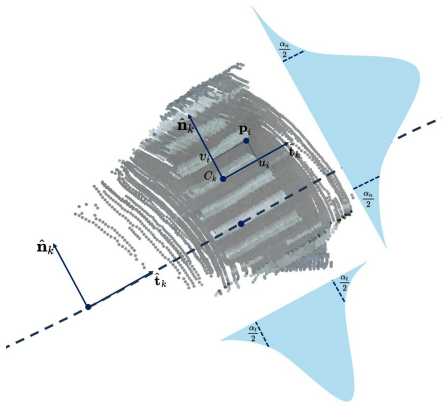
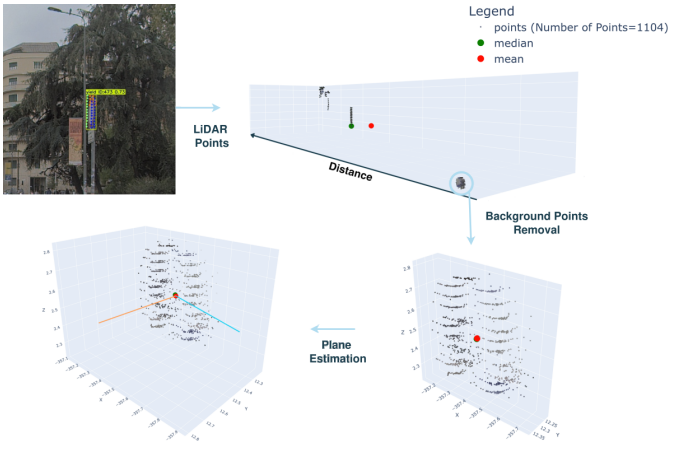


Fig. 3: Representation of quantile trimming to remove trapezoidal artifact in horizontal traffic sign point cloud.

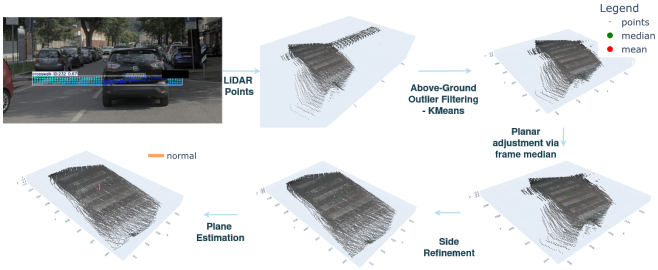
Finally, to retain only traffic signs that are relevant to the traversed route, the ego-path is exploited. Given the centroid of a sign and the closest point on the path, the sign is considered valid when the perpendicular distance between these two points is smaller than half of the sign's extent in the direction normal to the path; otherwise, it is discarded. Examples of this filtering process for both categories are shown in Fig. 4..

C. Semantic Path Annotation

The route annotation stage aims to convert the set of detected and localized traffic signs into a sequence of *behaviorally meaningful* annotations along the vehicle path. This task is non-trivial due to the presence of redundant information, such as advance signs, and the presence of multiple traffic signs that jointly encode a single driving constraint. Crucially, the annotation process prioritizes the *driving behavior* that the map is intended to enforce, rather than a purely geometric or visual representation of all observed road elements. At this stage of the pipeline, all horizontal and vertical traffic signs relevant to the vehicle's trajectory have been detected, tracked, localized, and filtered, resulting in a set denoted as $TS = \{ts_1, ts_2, \ldots, ts_N\}$ where each element represents a traffic sign instance applicable to the traversed route. The annotation process follows a hierarchical strategy that exploits the different levels of reliability and spatial accuracy associated with various types of traffic signs. Horizontal road markings are regarded as the most reliable source of spatial information, as they provide precise localization along the vehicle path and are not redundant. Vertical signs and additional panels are instead used to refine or disambiguate the semantic interpretation. The annotation procedure proceeds iteratively as follows:



(a) Vertical filtering



(b) Horizontal filtering

Fig. 4: Examples of the filtering process applied to detected traffic signs. (a) Vertical filtering. (b) Horizontal filtering.

1) **Anchor selection:** If horizontal markings are present in TS, one horizontal marking is selected as the anchor element. This marking defines the reference position along the path at which a special node may be placed.
2) **Redundant information aggregation:** All traffic signs in TS that are spatially close to the selected anchor are grouped together. Thresholds for aggregation are class dependent and are derived from the Italian Road Code [9]. This allows the merging of redundant evidence, such as vertical signs, horizontal text (e.g., "STOP"), or lane markings that convey the same behavioral constraint.
3) **Advance sign disambiguation:** To distinguish between advance signs and actual regulatory signs, an optical character recognition (OCR) pipeline [18] is applied to detect additional panels containing distance information. When such information is present, the search region for the corresponding actual sign is defined using the recognized distance, enlarged by a tolerance margin to account for localization uncertainty.
4) **Auxiliary semantic refinement:** In specific cases, additional semantic checks are performed. For instance, when a stop line is detected, an auxiliary detector [19] trained to recognize traffic lights is executed to determine whether the stop behavior is governed by a junction rule or by a traffic light, thereby refining the node type.
5) **Node annotation and confidence assignment:** Once all relevant information has been aggregated, the corresponding path node is annotated with the inferred semantic class and associated driving behavior. A confidence score is assigned based on the amount and consistency of the supporting evidence.
6) **Evidence removal and iteration:** All traffic signs that contributed to the current annotation are removed from TS. The process then repeats until TS is empty.

If no horizontal markings remain in TS, the annotation procedure is repeated by progressively relaxing the anchor selection criteria. First, horizontal text markings are considered as anchors. If these are also absent, vertical traffic signs without additional panels are used as the primary evidence. This hierarchical mechanism ensures that all relevant traffic signs contribute to the final semantic path annotation, even in the absence of ideal markings.

IV. Experiments

To assess the performance of the annotation method, 5 HD maps from our proprietary recordings were manually annotated and used to evaluate the process, consisting in more than 14 Km of Italian urban road. The objective is to evaluate the correctness of the placed nodes in the prediction vector $\hat{\mathbf{SN}}$ with respect to the ground truth vector $\mathbf{SN_{gt}}$. Each vector entry encodes the type of special node and its position (x, y) on the vehicle's path.

1) Matching Strategy and Metrics: Given every Predicted Node $\hat{n}$, Ground Truth Node n_{gt} and the polyline of the vehicle's path, it is possible to define metrics that vary across the allowable distance d:

- **True Positives** $TP(d)$: The number of ground truth nodes n_{gt} that have a matching predicted node $\hat{n}$ withing a certain signed distance d along the polyline of the path. Each ground truth node corresponds to at most one predicted node, the closest one.
- **False Negatives** $FN(d)$ The number of unmatched ground truths without a matching predicted node.
- **False Positives** $FP(d)$: The number of remaining unmatched predicted nodes, given a distance d, after removing from $\hat{\mathbf{SN}}$ the nodes that were correctly matched with a ground truth one.

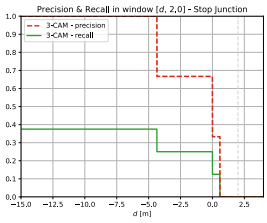
When a predicted node lies outside the admissible distance range for its class, the corresponding ground-truth node is counted as a false negative, and the predicted node, if present, is counted as a false positive. Each special node class is associated with a minimum and maximum admissible distance threshold, defining a class-specific validity window along the path. These thresholds account for both annotation variability and safety margin requirements. The adopted validity windows, together with the nominal placement distance relative to the physical point of interest (e.g., stop line or crosswalk center), are reported in Tab. I. The minimum distance threshold of $-7.0\,\mathrm{m}$ is applied uniformly across all classes. Negative distances indicate predictions placed before the ground truth

along the vehicle's path, while positive distances indicate placement after.

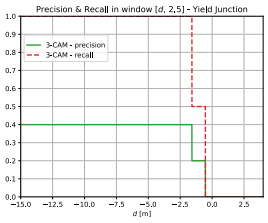
TABLE I: Distance thresholds for special node placement evaluation with respect to the global position of the actual element of interest.

Node Type	Validity window [Min, Max] (m)	Distance from Actual Element (m)	Window from Actual Element (m)
Stop at Junction	$[-7.0, 1.0]$	-2.0	$[-9.0, -1.0]$
Stop at Traffic Light	$[-7.0, 5.0]$	-6.0	$[-13.0, -1.0]$
Yield at Roundabout	$[-7.0, 2.5]$	-3.5	$[-10.5, -1.0]$
Yield at Crosswalk	$[-7.0, 5.0]$	-10.5	$[-17.5, -5.5]$
Yield at Junction	$[-7.0, 2.5]$	-3.5	$[-10.5, -1.0]$

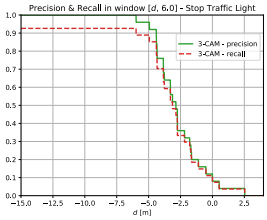
Based on these definitions, precision and recall are expressed as functions of the distance threshold d. For a given d, $P(d)$ and $R(d)$ are computed over the validity window $[d, \max_{\text{per class}}]$. This sliding-window formulation enables analysis of how performance varies as the tolerance for node placement accuracy is progressively relaxed. As shown in Fig. 5, both precision and recall remain high in the distance range $[-5\,\text{m}, 0\,\text{m}]$, indicating that the pipeline consistently places nodes either at the correct position or at distances that remain safe and behaviorally valid. In contrast, performance rapidly degrades for positive offsets, with precision and recall dropping toward zero within the range $(0\,\text{m}, 5\,\text{m}]$, and most curves reaching zero around $2.5\,\text{m}$. This behavior confirms that the pipeline rarely places nodes in regions that could lead to unsafe or risky vehicle actions.



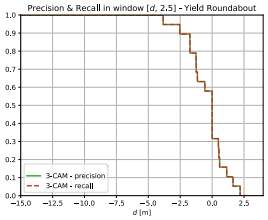
(a) Stop at Junction



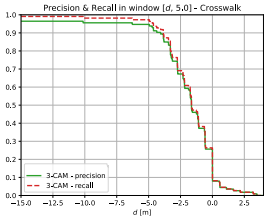
(b) Yield at Junction



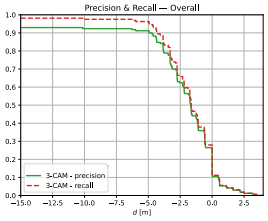
(c) Stop at Traffic Light



(d) Yield at Roundabout



(e) Yield at Crosswalk



(f) Overall

Fig. 5: Precision and recall curves as functions of distance threshold d for each special node class. The final subplot (f) presents the overall performance aggregating all classes.

2) Functional Equivalence Analysis: In addition to the strict class-matching metrics, a *functional equivalence analysis* is conducted. Unlike the standard evaluation, which penalizes any label mismatch, this analysis groups node classes that trigger identical vehicle behaviors. Specifically, if the system predicts a special node that differs from the ground truth but preserves the **correct safety-critical maneuver** (e.g., stopping or yielding), it is counted as a True Positive rather than a misclassification that would result in a FP-FN pair. This metric assesses the pipeline's robustness in guaranteed safe navigation, even in the presence of fine-grained semantic ambiguity. The functional groups defined for this analysis are:

- **Stop Maneuvers**: Aggregates *Stop at Junction* and *Stop at Traffic Lights*.
- **Yield Maneuvers**: Aggregates *Yield at Junction* and *Yield at Roundabout*.

Results for both the standard analysis and the functional equivalence analysis are presented in Tab. II.

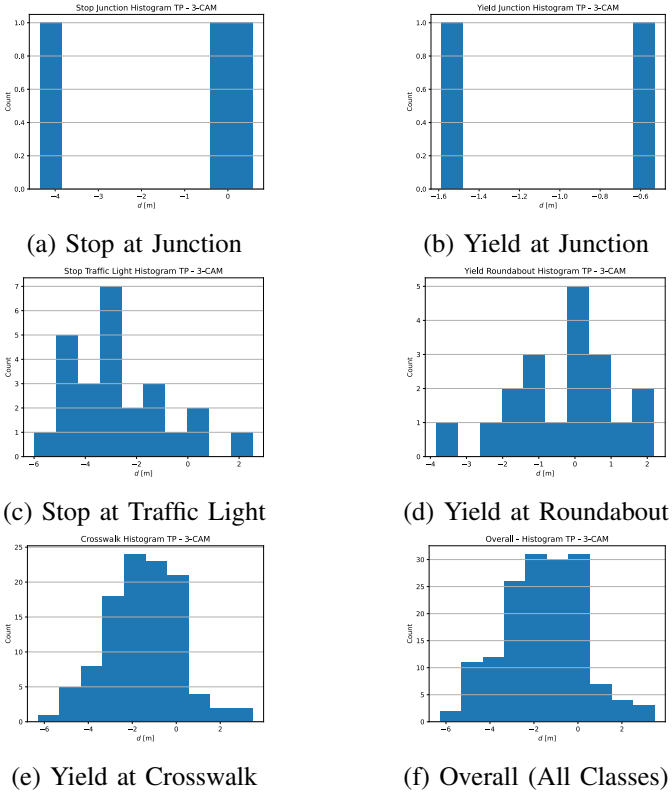
TABLE II: TP, FN, and FP for each special node class within their distance windows, comparing 3-CAM and 3-CAM-equivalent configurations.

Node Type	3-CAM			3-CAM-eq		
	TP	FN	FP	TP	FN	FP
Stop at Junction	3	0	5	3	0	5
Stop at Traffic Light	25	2	0	26	1	0
Yield at Roundabout	19	0	0	19	0	0
Yield at Crosswalk	108	2	5	108	2	5
Yield at Junction	2	0	3	2	0	3

The functional equivalence analysis reveals small but relevant improvements in critical metrics, meaning that few special nodes were misclassified due to the absence of informative signs and confirming that the underlying reasoning logic for node annotation is both sound and robust.

3) Distance Distribution of True Positives: To assess the spatial accuracy of node placement, we analyze the distribution of TP predictions relative to their corresponding ground truth positions. Fig. 6 presents histograms for each special node class, showing the distance offset between correctly predicted nodes and their ground truth counterparts within the respective validity windows defined in Tab. I. Across all classes, the distributions are skewed toward negative offsets, indicating a consistent tendency to place nodes conservatively before the ground-truth location, in line with the adopted safety-aware design.

The evaluation of the pipeline according to these metrics demonstrates consistent performance in both positional accuracy and behaviorally equivalent annotation. The motivation for adopting this evaluation strategy arises from several limitations that affect both the assessment process and the interpretation of the results. In particular, the absence of a unified convention for manual node placement along the vehicle path within the annotation workflow introduces unavoidable variability in the ground-truth data. This variability originates from three main sources. First, **discretization error** introduces a baseline positional uncertainty, since nodes can only be placed at discrete intervals of $0.5,\text{m}$. Second, **operator variability** leads to slight differences in node placement, as individual



(a) Stop at Junction (b) Yield at Junction

(c) Stop at Traffic Light (d) Yield at Roundabout

(e) Yield at Crosswalk (f) Overall (All Classes)

Fig. 6: Distance distribution of TP predictions relative to ground truth for each special node class. Each histogram is computed within the class-specific validity window. Subplot (f) shows the aggregated distribution across all node classes.

annotators may interpret the correct location differently based on experience and judgment. Third, **visual estimation limitations** arise because operators must infer distances solely from camera imagery, making precise placement inherently difficult.To account for these factors, the proposed evaluation metrics explicitly incorporate asymmetric distance tolerances and safety-aware validity windows.

V. Conclusions

This paper presented a fully automated offline pipeline for the semantic annotation of navigation paths for autonomous driving, with the goal of reducing effort, subjectivity of manual path annotation while increasing scalability. The system combines camera-based detection with LiDAR-based depth estimation to robustly localize both vertical traffic signs and horizontal road markings. This information is temporally aggregated to exploit the complete traversal history to filter spurious detections, refine spatial estimates, and resolve ambiguities. Furthermore, the hierarchical annotation logic enables the consistent fusion of heterogeneous and redundant regulatory cues into behaviorally meaningful path annotations. Experimental validation on more than 14 km of Italian urban roads demonstrates that the proposed pipeline achieves high annotation accuracy and strong geometric consistency, with the majority of remaining errors attributable to particularly challenging visual conditions, such as worn markings or low-contrast pavement. The safety-aware evaluation methodology and the functional equivalence analysis further show that, even in the presence of limited semantic ambiguity, the system reliably preserves the correct driving behavior, which is the primary requirement for safe autonomous navigation.

References

[1] C. Merolla, S. Specchia, A. Galimberti, M. Corno, G. Panzani, and S. M. Savaresi, "Autonomous vehicle intersection management through speed optimization and finite state machine-based decision making," in *Proceedings of the 2025 IEEE International Conference on Intelligent Transportation Systems (ITSC), Gold Coast, Australia, November 18-21*, 2025.

[2] Y. Shen and J. Li, "Utilizing navigation paths to generate target points for enhanced end-to-end autonomous driving planning," 2024.

[3] X. Xiao, S. Yang, M. Fan, S. Xu, X. Liu, W. Hu, and H. Xiong, "Towards v2x hd mapping for autonomous driving: A concise review," *2025 IEEE Intelligent Vehicles Symposium (IV)*, pp. 998–1005, 2025.

[4] B. Liao, S. Chen, X. Wang, T. Cheng, Q. Zhang, W. Liu, and C. Huang, "Maptr: Structured modeling and learning for online vectorized hd map construction," *ArXiv*, vol. abs/2208.14437, 2022.

[5] T. Yuan, Y. Liu, Y. Wang, Y. Wang, and H. Zhao, "Streammapnet: Streaming mapping network for vectorized online hd map construction," *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 7341–7350, 2023.

[6] S. Kunsági-Máté, L. Peto, L. Seres, and T. Matuszka, "Accurate automatic 3d annotation of traffic lights and signs for autonomous driving," *ArXiv*, vol. abs/2409.12620, 2024.

[7] M. S. Alam, A. Sani, M. T. Islam, A. Ghosh, and A. Chowdhury, "Real-time traffic sign detection for autonomous vehicles using yolov11," pp. 1–6, 02 2025.

[8] H. Lyu, J. S. Berrio Perez, Y. Huang, K. Li, M. Shan, and S. Worrall, "Online high-definition map construction for autonomous vehicles: A comprehensive survey," *Journal of Sensor and Actuator Networks*, vol. 14, no. 1, 2025.

[9] Repubblica Italiana, "Codice della strada," 2023. Decreto Legislativo 30 aprile 1992, n. 285, consolidated version in force.

[10] D. Tabernik and D. Skočaj, "Deep Learning for Large-Scale Traffic-Sign Detection and Recognition," *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[11] C. Ertler, J. Mislej, T. Ollmann, L. Porzi, G. Neuhold, and Y. Kuang, "The Mapillary Traffic Sign Dataset for Detection and Classification on a Global Scale," May 2020. Version 2.

[12] R. Timofte, K. Zimmermann, and L. Van Gool, "Multi-view Traffic Sign Detection, Recognition, and 3D Localisation," in *Proceedings of the IEEE Workshop on Applications of Computer Vision (WACV)*, pp. 1–8, Dec. 2009.

[13] F. Larsson and M. Felsberg, "Using Fourier Descriptors and Spatial Models for Traffic Sign Recognition," in *Proceedings of the Scandinavian Conference on Image Analysis (SCIA)*, vol. 6688, pp. 238–249, May 2011.

[14] R. Oprea, "Traffic signs detection europe dataset," Feb. 2024. Accessed: 2025-10-10.

[15] O. Detection, "Polish traffic signs detection dataset." https://universe.roboflow.com/object-detection-exlbj/polish-traffic-signs-detection, dec 2024. visited on 2025-10-19.

[16] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "BoT-SORT: Robust Associations Multi-Pedestrian Tracking," *arXiv preprint arXiv:2206.14651*, July 2022.

[17] F. Zhang, J. Zhang, Z. Xu, J. Tang, P. Jiang, and R. Zhong, "Extracting traffic signage by combining point clouds and images," *Sensors*, vol. 23, no. 4, 2023.

[18] C. Cui, T. Sun, M. Lin, T. Gao, Y. Zhang, J. Liu, X. Wang, Z. Zhang, C. Zhou, H. Liu, Y. Zhang, W. Lv, K. Huang, Y. Zhang, J. Zhang, J. Zhang, Y. Liu, D. Yu, and Y. Ma, "Paddleocr 3.0 technical report," 2025.

[19] O. Belotti, A. De Luca, R. Pieroni, M. Corno, and S. M. Savaresi, "Camera-lidar traffic light relevance estimation as graph-matching for autonomous vehicles," in *Proceedings of the 2025 IEEE International Conference on Intelligent Transportation Systems (ITSC), Gold Coast, Australia, November 18-21*, 2025.