Semantic Map Learning of Traffic Light to Lane Assignment based on Motion Data

Thomas Monninger[1], Andreas Weber[2], Steffen Staab[3,4]

***Abstract*— Understanding which traffic light controls which lane is crucial to navigate intersections safely. Autonomous vehicles commonly rely on High Definition (HD) maps that contain information about the assignment of traffic lights to lanes. The manual provisioning of this information is tedious, expensive, and not scalable. To remedy these issues, our novel approach derives the assignments from traffic light states and the corresponding motion patterns of vehicle traffic. This works in an automated way and independently of the geometric arrangement. We show the effectiveness of basic statistical approaches for this task by implementing and evaluating a pattern-based contribution method. In addition, our novel rejection method includes accompanying safety considerations by leveraging statistical hypothesis testing. Finally, we propose a dataset transformation to re-purpose available motion prediction datasets for semantic map learning. Our publicly available API for the Lyft Level 5 dataset enables researchers to develop and evaluate their own approaches[5].**

I. Introduction

Autonomous vehicles require a semantic understanding of the given traffic scene to navigate complex environments safely. At intersections, understanding the assignment of traffic lights to lanes is a prerequisite to determining whether to stop. This assignment information is used in safety-critical applications and currently cannot be derived by an online system with the required reliability.

The traffic light to lane assignment (TL2LA) is defined by the geometric arrangement of traffic lights relative to the lanes in an intersection and optionally by indication inlays inside the traffic light bulbs, such as arrows. Using this information to automate the map annotation in a scalable way does not reach the required level of correctness. The vast variety of geometric configurations of traffic lights and lanes and the unreliable detection of traffic light inlays make it very challenging to precisely assign individual traffic lights to their respective lanes. Hence, the assignment is traditionally provided a priori from an HD map, which involves laborious manual annotation efforts. Modeling the broad varieties of intersection branches, topology, arrangement of traffic lights, etc., constitutes its own modeling and knowledge acquisition problem [1]. Fully capturing the geometric layout and semantic relationships between traffic elements of an urban traffic scene may be difficult even for humans.

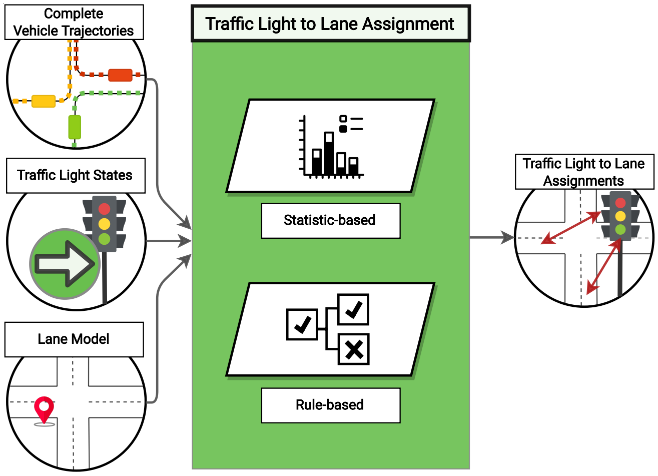
Complete Vehicle Trajectories
Traffic Light States
Lane Model
Traffic Light to Lane Assignment
Statistic-based
Rule-based
Traffic Light to Lane Assignments

Fig. 1. Overview of our approach to derive traffic light to lane assignments from detected traffic light states and motion patterns of vehicle traffic.

Current research has rarely considered the learning of semantic map information. One reason is that systems with high level of autonomy (e.g., robotaxis) use an HD map as a strong prior to meet the essential requirement for high safety. On a limited scale, HD maps can be precisely annotated by humans, dropping the need for automated solutions. Secondly, the limited prior work has focused on deriving TL2LA from the geometric arrangement of traffic lights and lanes at intersections. These approaches require labeled data for a multitude of intersection layouts to generalize well.

In this paper, we propose a novel paradigm of learning semantic map features from motion patterns of vehicle traffic. Our main observation is that humans resolve the TL2LA problem while driving, leading to motion patterns that implicitly contain the required data. We implement two methods for statistical and rule-based discovery of the TL2LA from detected traffic light states and the respective motion patterns as shown in Figure 1. This approach is independent of the geometric arrangement and therefore generalizes to any country and intersection layout. Finally, we present a dataset transformation to re-purpose available datasets for this task and evaluate the two proposed methods.

In summary, our main contributions are:

- We propose a novel learning task, deriving the TL2LA from motion patterns of traffic, and provide two methods to solve this task including safety considerations.
- We present a dataset transformation that changes the

[1] Mercedes-Benz Research & Development North America, Sunnyvale, CA, USA (email: thomas.monninger@mercedes-benz.com)

[2] Mercedes-Benz AG, Research & Development, Stuttgart, Germany (email: andreas_silvius.weber@mercedes-benz.com)

[3] University of Stuttgart, Institute of Parallel and Distributed Systems, Stuttgart, Germany (email: steffen.staab@ipvs.uni-stuttgart.de)

[4] University of Southampton, Electronics and Computer Science, Southampton, United Kingdom

[5] https://github.com/tmonnin/semantic_map_learning

representation of motion prediction datasets to make them suitable for learning map semantics.

- We evaluate our methods on the transformed Lyft Level 5 dataset and provide an API for future research in the field of motion-based semantic map learning.

This work is structured as follows: Section II discusses related work. Section III presents problem domain, problem statement and our approach. Implementation and experiments are described in sections IV and V. Finally, section VI covers remaining limitations and section VII gives a conclusion.

II. Related Work

This section reviews related work concerning semantic map learning, deriving information from motion data and available datasets.

A. Semantic Map Learning

Maps comprise geometric objects and semantic features, which are both needed to perform the driving task. Geometric objects, such as signs and crosswalks, are directly perceivable by sensors. Semantic map features are virtual and relate to geometric objects in the real world. Those include derived entities such as lanes, which model the lane corridor and are constrained by the geometric lane dividers. Another type of semantic map features are relationships between map entities, i.e., the TL2LA.

One way of deriving the TL2LA is by applying heuristics to the geometric arrangement. Early work from Fairfield and Urmson [6] uses a simple heuristic to add the TL2LA to a geometric map of the traffic lights. They acknowledge the challenge of this task and integrate a human verification step in their process because the heuristic alone did not provide the desired quality. Poggenhans [7] defines a more methodical approach by developing a rule set from the official traffic regulations as a heuristic. Similarly, that approach derives a geometric arrangement of the road infrastructure first. In a second step, the proposed rule set is used to solve semantic relations such as right of way and TL2LA. Again, the concluding statement indicates that heuristic approaches are insufficient because they rely on correct and unambiguous infrastructure, which is not always given.

Alternative ways of deriving TL2LA include learning-based approaches on the geometric arrangement. Li et al. [8] formulate the problem of semantic map learning and predict semantic map geometries in bird's-eye view from sensor data. Similar to other works [9], [10], [11], they fall short of deriving semantic relations like the TL2LA. Langenberg et al. [12] target the TL2LA in image space. They transform the lower part of the camera image using an inverse perspective mapping to get a top-down view of the road surface and use a learning-based model to predict the TL2LA directly from that image.

All approaches listed above tackle the problem of semantic map learning by deriving semantics from the geometric representation. This requires a diverse dataset to cover all variations of the real world. In contrast, our approach does not use geometry information of traffic lights, i.e., their position and heading are not input to the approach.

B. Deriving Information from Motion Data

Prior work has used motion data in the form of vehicle trajectories to derive information.

There is substantial literature on general clustering and pattern recognition on motion data. Yuan et al. [13] review moving object trajectory clustering algorithms. While they do not target semantic map learning, the methods described are a precondition of deriving information from massive motion data. Jiao et al. [14] show an approach for characterizing motion data based on summarizing and classifying patterns. These works do not attempt TL2LA but provide general ideas for working with motion data.

Scientific work in the context of mapping from motion data mostly targets geometric map features. Early work from Chen and Krumm [15] and from Uduwaragoda et al. [16] use statistical models to predict traffic lanes from GPS traces. More recent work aims to close the gap to HD maps by deriving additional geometric map features such as boundaries and signs [17] by using higher-level features such as lane marking types provided by the vehicle fleet [18] or by directly extracting lane-level information from raw motion patterns [19].

Only a few publications describe how to derive semantic map attributes from motion data. Derrow-Pinion et al. [20] predict the estimated time of arrival for a queried map route using motion data transmitted from mobile devices. Wirthmüller et al. [21] model a semantic lane attribute, the probability of lane change events, based on data from the vehicle fleet. Our novel approach aims to derive the TL2LA purely from motion data, a topic that has not yet been explored in the literature.

TABLE I

Comparison of the dataset meta information provided by different motion prediction datasets.

Meta Information	**Argoverse 2.0 [2]**	**nuScenes [3]**	**Waymo Motion [4]**	**Lyft Level 5 [5]**	**Requirements Map Learning**
Number of scenes N	250 000	41 000	103 000	162 000	As high as possible
Scene duration τ	11 s	20 s	20 s	25 s	At least multiple seconds
Unique roadways R	2110 km	4300 km	1750 km	10 km	High as long as density high
Spatial recording density ρ	$1303\,\frac{\mathrm{s}}{\mathrm{km}}$	$190\,\frac{\mathrm{s}}{\mathrm{km}}$	$529\,\frac{\mathrm{s}}{\mathrm{km}}$	$405\,000\,\frac{\mathrm{s}}{\mathrm{km}}$	As high as possible
Total time	763 h	228 h	570 h	1125 h	As high as possible
Size	58 GB	48 GB	1.4 TB	78 GB	
Coordinate System	global	global	local	global	Global to localize scenes
License	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0	CC BY-NC-SA 4.0	

C. Available Datasets

High-quality, large-scale datasets are crucial to train models for autonomous driving, especially in the mapping domain with extremely diverse real-world scenarios and highly manual labeling efforts. Unfortunately, there are no publicly available datasets that specifically address the problem of semantic map learning. However, there are several public motion prediction datasets that address the primary goal of motion forecasting in urban environments [2], [3], [22], [4], [5]. They contain time sequences of motion data, and some additionally include HD maps with semantic annotations for improved motion prediction.

Table I lists publicly available motion prediction datasets and gives an overview of their metadata. The rightmost column addresses requirements for semantic map learning, which are discussed in our approach.

III. Motion-based Approach to derive TL2LA

This section formalizes the problem domain and problem statement of deriving the TL2LA based on motion patterns. We propose two methods to solve the problem and explain safety considerations.

A. Formalizing the Problem Domain

Let V be the set of all vehicles and $\mathrm{pos} : V \times T \to \mathbb{R}^2$ be a partial function that maps each vehicle to its position at time $t \in T$. Kinematics are given by the functions $\mathrm{vel}(v,t) = \partial \mathrm{pos}(v,t)/\partial t$ for velocity and $\mathrm{acc}(v,t) = \partial \mathrm{vel}(v,t)/\partial t$ for acceleration of vehicle v at time t. Let S be the set of all traffic lights and $\mathrm{state} : S \times T \to \{\mathrm{red}, \mathrm{green}\}$ be a partial function that maps each traffic light to its state (red or green) at time $t \in T$. Let L be the set of all lanes on the road and $\mathrm{boundary}(l) = (b_l, b_r)$ with $b \in \mathbb{R}^{2k} \times \mathbb{R}^{2k}$ be a function that maps each lane to its left and right boundary b as a sequence of k points in $\mathbb{R}^2$. The TL2LAs are represented by a function $\mathrm{assign} : S \times L \to \{0, 1\}$, which maps each pair of traffic light and lane to its binary value (0 for no assignment, 1 for assignment).

B. Problem Statement

The required input data for deriving TL2LA from motion data comprises the lane geometry ($\mathrm{boundary}$), the position of vehicles over time (pos), and the traffic light state over time (state). Lane topology and the mapping of vehicles to lanes are optional, because these can be derived geometrically.

We assume to have a dataset:

$$D = (V, S, L, \mathrm{pos}, \mathrm{state}, \mathrm{boundary}) \tag{1}$$

Predicting the TL2LA is a binary classification problem. Let target output Y be the TL2LA for each pair of traffic light and lane, as represented by the definition of the function assign:

$$Y = \mathrm{assign}(s, l) \tag{2}$$

Then the goal is to find a function:

$$g\left(V, s, l, \mathrm{pos}, \mathrm{state}, \mathrm{boundary}\right) = Y = \mathrm{assign}(s, l) \tag{3}$$

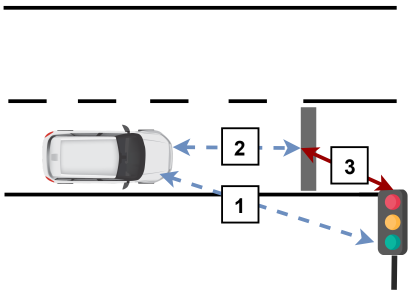

Fig. 2. Steps to derive TL2LA (3) based on the traffic light to vehicle (1) and vehicle to lane assignment (2).

C. Methods to derive TL2LA

To get from temporal correspondences between motion patterns of vehicles and traffic light states to TL2LAs, the following procedure is needed as shown in Figure 2:

1) Based on motion pattern $\mathrm{pos}(v,t)\,\mathrm{for}\, t \in T$ of vehicle v and the state over time $\mathrm{state}(s,t)\,\mathrm{for}\, t \in T$ of traffic light s: Derive evidence for or against an assignment between v and s.
2) Based on geometric position $\mathrm{pos}(v,t)$ and lane geometries $\mathrm{boundary}(l)\,\mathrm{for}\, l \in L$: Use geometric matching $\mathrm{loc} : V \times T \to L$ to assign vehicle v to its driving lane l at time t.
3) Determine the TL2LA $\mathrm{assign}(s,l)$ between traffic light s and lane l by aggregating individual evidences predicting the relation between l and s.

We propose two statistical methods and compare with a naive baseline.

1) Naive Baseline Method: Predict the class with the highest prior probability, which can be formalized as:

$$\mathrm{assign}_{\mathrm{prior}}(s, l) = \arg\max_{c_i \in C} n_i \tag{4}$$

where $C = \{0,\ 1\}$ is the set of TL2LA classes (0 for no assignment, 1 for assignment) and n_i is the number of occurrences of class c_i in the dataset.

2) Pattern-based Contribution Method: This method extracts motion patterns individually from each point in time and weighs the contributions w.r.t. the TL2LAs. A heuristic function $h : V \times S \times T \times \mathrm{Cond} \to \mathbb{R}$ calculates a contribution value given a vehicle, a traffic light state, a point in time and Cond, a set of condition functions defined below. The heuristic function h is defined such that positive values contribute to an assignment and negative values have the opposite effect. The decision about an assignment is made by summing up all individual contribution values that were generated by the heuristic function:

$$\mathrm{assign}_{\mathrm{pattern}}(s, l) = \sum_{t \in T} \sum_{(v,s) \in V \times S} h\left(v, s, t, \mathrm{cond}\right) \tag{5}$$

Table II shows the definition of h with the set of condition functions $\mathrm{cond} = \{\mathrm{vel},\ \mathrm{acc},\ \mathrm{distance},\ \mathrm{is\_lead},\ \mathrm{turn\_type},\ \mathrm{state\_duration}\}$. Those are a) the distance of a vehicle to the intersection $\mathrm{distance} : V \times T \to \mathbb{R}$ with two thresholds $\mathrm{stop\_zone} = 8\,\mathrm{m}$ and $\mathrm{slow\_zone} = 20\,\mathrm{m}$,

TABLE II

HEURISTIC FUNCTION h BASED ON THE VELOCITY (vel) AND ACCELERATION (acc) OF VEHICLE AND TRAFFIC LIGHT (TL) STATE.

Pattern	**Kinematics**	**TL State**	**Additional Conditions**	**Heuristic** h
Stationary	$\lvert\text{vel}\rvert < 1 \wedge$ $\lvert\text{acc}\rvert < 1$	red	$(\text{distance} < \text{stop\_zone})$	$+2$
		green	is_lead	-1
			$\text{is\_lead} \wedge (\text{state\_duration} > \text{reaction\_time\_green})$	-3
Continuously moving	$\lvert\text{vel}\rvert \geq 1 \wedge$ $\lvert\text{acc}\rvert < 1$	red	$(\text{distance} < \text{slow\_zone})$	-1
			$(\text{distance} < \text{stop\_zone}) \wedge (\text{state\_duration} > \text{reaction\_time\_red})$	-3
		green	$(\text{distance} < \text{slow\_zone})$	$+3$
			$(\text{distance} < \text{stop\_zone})$	$+5$
Acceleration from stationary	$\lvert\text{vel}\rvert < 1 \wedge$ $\lvert\text{acc}\rvert \geq 1 \wedge$ $\text{acc} > 0$	red	$(\text{distance} < \text{stop\_zone}) \wedge (\text{turn\_type} \neq \text{right\_turn})$	-2
		green	$(\text{distance} < \text{slow\_zone}) \wedge$ $(\text{state\_duration} > \text{reaction\_time\_red})$	$+3$
Acceleration while moving	$\lvert\text{vel}\rvert \geq 1 \wedge$ $\lvert\text{acc}\rvert \geq 1 \wedge$ $\text{acc} > 0$	red	$(\text{distance} < \text{slow\_zone})$	-1
			$(\text{distance} < \text{stop\_zone}) \wedge (\text{state\_duration} > \text{reaction\_time\_red})$	-3
		green	$(\text{distance} < \text{slow\_zone})$	$+1$
Deceleration	$\lvert\text{vel}\rvert \geq 1 \wedge$ $\lvert\text{acc}\rvert \geq 1 \wedge$ $\text{acc} < 0$	red	$(\text{distance} < \text{slow\_zone})$	$+2$
		green	$(\text{distance} < \text{stop\_zone}) \wedge \text{is\_lead} \wedge (\text{turn\_type} = \text{left\_turn})$	-1
			$(\text{distance} < \text{stop\_zone}) \wedge \text{is\_lead} \wedge (\text{turn\_type} = \text{straight})$	-2
Other	else	red/green		0

b) whether the vehicle is the first vehicle before the intersection entry $\text{is\_lead} : V \times T \rightarrow \{\text{true}, \text{false}\}$, c) the turn information of the current lane $\text{turn\_type} : L \rightarrow \{\text{left\_turn}, \text{right\_turn}, \text{straight}\}$, d) the duration of the current traffic light state $\text{state\_duration} : S \times T \rightarrow \mathbb{R}$ relative to the time delay of a vehicle reacting to a traffic light state change ($\text{reaction\_time\_red} = 1\,\text{s}$ and $\text{reaction\_time\_green} = 3\,\text{s}$). By applying this set of rules, contribution values for a pair of traffic light and lane are calculated for each point in time. When the aggregated value of all contributions for a pair of traffic light and lane is positive, the method predicts $\text{assign}(s, l) = 1$, otherwise it predicts $\text{assign}(s, l) = 0$. Predictions are aggregated over all points in time based on the majority class.

3) Rejection Method: The rejection method formulates the problem as a hypothesis test. Conservatively assume an assignment for all pairs of traffic lights and lanes (H_0): $\forall (s, l) \in S \times L : \text{assign}(s, l) = 1$. Set $\text{assign}(s, l) = 0$ only if a significant number of vehicles have been recorded passing the intersection on lane l while $\text{state}(s, t) = \text{red}$.

$$\begin{aligned} H_0 :&\quad \text{assign}(s, l) = 1 \quad \text{(assignment)} \\ H_1 :&\quad \text{assign}(s, l) = 0 \quad \text{(no assignment)} \end{aligned}$$

The detection of a vehicle passing the intersection is defined relative to the intersection entry. If the velocity of a vehicle directly in front of the intersection entry ($\text{distance} < 1\,\text{m}$) is greater than a specific threshold ($\text{vel} > 15\,\frac{\text{km}}{\text{h}}$), a pass is assumed.

A binomial hypothesis test is used to reject the null hypothesis based on a significance level. By initially assuming a true assignment, this method minimizes false negatives and optimizes for recall. A further advantage is that this method provides an output for all pairs of traffic lights and lanes. We assume that a TL2LA exists if less than 5 % of recorded vehicles pass on a red light (binomial distribution with $p = 0.05$). To minimize the likelihood of false rejections of H_0, we choose a significance level $\alpha = 0.001$. For a pair of traffic light s and lane l, let n be the overall number of passes in the dataset where $\text{loc}(v, t) = l$ for $v \in V, t \in T$ while the traffic light s is detected with $\text{state}(s, t)$ and let k be the number of the subset of passes with $\text{state} = \text{red}$. Then the TL2LA is derived as:

$$\text{assign}_{\text{rejection}}(s, l) = [\text{Binomialtest}(k, n, p) < \alpha] \qquad (6)$$

The concept of right turn on red contradicts the traffic light to lane assignment, since vehicles are allowed to pass the red light after stopping for crossing traffic. As a special heuristic, our rejection method only invalidates the assignment of a traffic light to a right-turning lane for passes with a significantly higher velocity ($v > 25\,\frac{\text{km}}{\text{h}}$). Unprotected turns are handled implicitly by the rejection method, since it does not extract evidence from stopping vehicles, but purely invalidates TL2LAs given passes on red light.

D. Safety Considerations

To ensure maximum safety, an autonomous vehicle should not pass an intersection entry when any of the assigned traffic lights is red. Therefore, false negative TL2LAs are critical, since those might result in ignoring the relevant traffic light. A false positive TL2LA results in regarding an additional traffic light. This might result in stopping at an actual green light, which is less critical from a safety perspective. We design our proposed methods to account for the adjusted Bayes risk. In our pattern-based contribution method, we design h in favor of a higher recall. In our rejection method, we assume $\text{assign}(s, l) = 1$ as the null hypothesis. It is important to highlight that our proposed methods do not cover all edge cases and are not suitable for deployment (see Section VI for limitations).

In addition to predicting the assignment, it is essential to assess the confidence of the output. From an information-theoretical perspective, confidence is based on how many different combinations of traffic light states have been observed, and the consistency between traffic light states and motion patterns given the derived TL2LAs. By the law of large numbers, the likelihood of seeing all state combinations rises with an increasing amount of samples and the accuracy runs into saturation. Our rejection method uses a hypothesis test that provides a p-value as a function of the number of samples and the consistency. Therefore, it can be used as a direct measure of confidence.

IV. Implementation on Dataset

This section proposes the use of widely available motion prediction datasets for semantic map learning. Motion prediction datasets are compared regarding their use for deriving the TL2LA. We perform a dataset transformation on the Lyft Level 5 dataset [5] and explain data preparation steps.

A. Motion Prediction Datasets for Semantic Map Learning

The problem of motion prediction is defined by a function:

$$\text{pred}\left(V, S, L, \text{pos}_{0:t}, \text{state}, \text{boundary}, \text{assign}\right) = \text{pos}_{t+1:T} \tag{7}$$

It predicts future trajectories $\text{pos}_{t+1:T}$ from past trajectories $\text{pos}_{0:t}$. Additional inputs often include lanes (L) with boundaries (boundary), traffic lights S with their states (state), and the TL2LAs (assign). We show that specific datasets for motion prediction can be transformed into datasets for semantic map learning. A comparison with function g for the TL2LA problem (c.f., Equation 3) shows the required steps to transform the dataset (also visualized in Figure 3):

1) Future trajectories are appended to past trajectories to form complete vehicle trajectories as input: $\text{pos} = \text{pos}_{0:T} = \left[\text{pos}_{0:t}, \text{pos}_{t+1:T}\right]$
2) Semantic relations, specifically TL2LAs, are used as output instead of input: $Y = \text{assign}$

Datasets for motion prediction are usually annotated from recordings of a measurement vehicle (referred to as ego). The partial functions state and pos are only defined for traffic lights and vehicles in the field of view of the measurement vehicle. We can use this locality to formulate a condition for a pair of traffic light s and lane l:

$$\exists t \in T : \text{state}(s, t) \land \text{pos}(v, t) \land (\text{loc}(v, t) = l) \tag{8}$$

It requires that there exists a point in time in the dataset where a state for s and position for a vehicle v are detected while v is on l. This condition is not met for most pairs in motion prediction datasets, since most pairs of traffic lights and lanes are not part of the same intersection. Predicting a TL2LA is not useful in those cases where there is no evidence of motion patterns on that lane relative to the traffic light state.

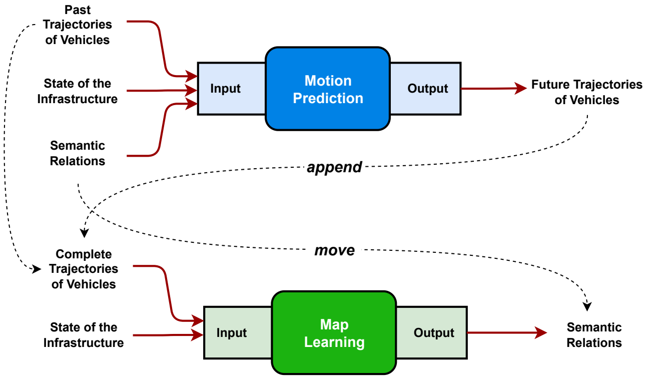
Past Trajectories of Vehicles
State of the Infrastructure
Semantic Relations
Input
Motion Prediction
Output
Future Trajectories of Vehicles
append
move
Complete Trajectories of Vehicles
State of the Infrastructure
Input
Map Learning
Output
Semantic Relations

Fig. 3. Visualization of the dataset transformation that converts motion prediction datasets into a representation suitable for learning map semantics from motion data. Semantic relations are used as outputs instead of inputs.

TABLE III
Comparison of relevant dataset features

Feature	Argoverse 2	nuScene	Waymo	Lyft
Lane geometry	✓	✓	✓	✓
Lane topology	✓	✓	✓	✓
Vehicle trajectory	✓	✓	✓	✓
Vehicle to lane mapping	✗	✗	✗	✗
Traffic light state	✗	✓	✓	✓
Traffic light geometry	✗	✗	✗	✓
TL2LA	✗	✗	✓	✓

B. Comparison of available Datasets

Table III shows which available motion prediction datasets exhibit which features. Only the datasets from Waymo and Lyft allow us to derive TL2LA from motion data.

To perform statistics on the motion patterns relative to a traffic light state, a suitable dataset needs to have many recordings of the same intersection. The datasets in Table I do not provide information at the intersection level. Instead, we provide a different metric by defining a spatial recording density ρ as follows:

$$\rho = \frac{N \cdot \tau}{R} \tag{9}$$

where N is the number of scenes, τ is the duration of a scene, and R is the length of unique roadways. Table I shows the resulting values. The Lyft dataset has many scenes on a small set of unique roadways. Hence, the resulting spatial recording density is high at $405\,000\,\frac{\text{s}}{\text{km}}$. Its annotated map contains 12 intersections with 22 annotated intersection branches and 279 TL2LAs in total.

Figure 4 visualizes the semantic map of the Lyft dataset in Palo Alto, California. The color represents the number of scenes recorded by a fleet of 20 vehicles. The heatmap for the zoomed-in intersection *y4Ss* shows that the ego vehicle does not travel all lanes. Therefore, using only the motion patterns of the ego vehicle will not give information for all TL2LAs in the semantic map. In addition, the TL2LAs in the Lyft dataset are only partially annotated to cover intersection branches traveled by the ego vehicle. Another geospatial requirement is that a suitable dataset should provide the map in a global

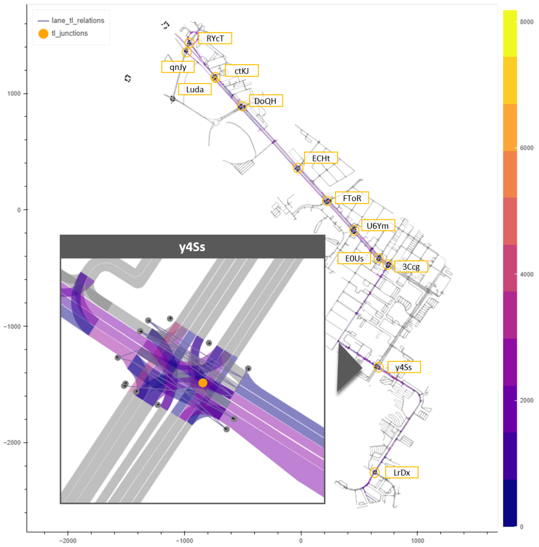

Fig. 4. Visualization of the semantic map of the Lyft dataset, including lane segments, traffic light geometries and TL2LAs. The heatmap indicates how often the ego vehicle traveled a specific lane segment.

coordinate system to avoid the need to localize the scenes to each other. This is also one evaluation criterion listed in Table I and fulfilled by Argoverse 2, nuScenes and Lyft datasets. In summary, the Lyft dataset best meets all criteria and is used in the further course of this work.

C. Dataset Preparation

Making use of the Lyft dataset for map learning requires some additional preprocessing steps. Also, the vast amount of available scenes allows removing data points that provide no clear evidence, e.g., due to noise. The following steps are performed:

- Lanes are represented in the Lyft dataset as segments. We manually label lanes by forming sequences of lane segments from the intersection entry $20\,\mathrm{m}$ backward. This also filters out overlapping lane segments and asserts unique geometric assignments from vehicle to lane by loc).
- Only vehicles on the same intersection branch and at a certain distance to the intersection are considered. Those are close to the intersection entry, so their motion pattern is highly related to the traffic light state.
- The TL2LA label is only kept for those pairs of traffic lights and lanes that meet the condition stated in Equation 8. This way, pairs not part of the same intersection are filtered out and 279 pairs remain in the Lyft dataset.

V. Experiments

We evaluate our methods on two subsets of the Lyft dataset, considering 1.) only trajectories of the recording vehicle, referred to as ego vehicle and 2.) all vehicle trajectories. Considering only ego trajectories yields reliable motion patterns and avoids tracking and occlusion issues. However, this limits the predictable set of TL2LAs, since not all lanes of each intersection were traveled by the ego vehicle in the Lyft dataset.

A. Metrics

Accuracy (Acc), precision (Prec), recall and F-Score (F_1) are evaluated. Recall is most important, because false negative TL2LAs are most critical for safety (see Section III-D). Precision is still essential, since false positive TL2LAs might yield unnecessary stops at actual green light.

B. Quantitative Results

Table IV reports the quantitative results of the three methods on the Lyft dataset. The rejection method can only classify 271 out of the 279 pairs of traffic lights and lanes due to a limitation of the Lyft dataset. All methods were evaluated on that set of 271 pairs to have a direct comparison.

1) Naive Baseline Method: The Lyft dataset consists of 109k scenes at traffic light-controlled intersections that include detections of traffic light states. The naive baseline method can use all 109k scenes and returns the most probable TL2LA class in the Lyft dataset, which is $\mathrm{assign}(s, l) = 1$ for every pair of traffic light and lane. Therefore, it reaches 100 % recall, as well as 90.0 % F_1 score on ego trajectories only and 79.8 % F_1 score on all vehicles.

2) Pattern-based Contribution Method: The pattern-based contribution method only uses scenes within a certain distance to the intersection. From the 109k scenes, 90k are available when considering all vehicles and 64k are available when considering only the ego vehicle.

The pattern-based contribution method roughly matches the performance of the naive baseline method on ego trajectories only, but outperforms it on all vehicles with an 81.8 % vs. 79.8 % F_1 score. This indicates that the pattern-based contribution can exploit evidence from the motion patterns of vehicles to verify or falsify a TL2LA. The comparison with our rejection method shows a lower performance of the pattern-based contribution method considering their accuracy, recall, and F_1 scores. Only its precision is considerably higher with 83.2 % vs. the precision of the rejection method with 78.3 %. We believe that a more sophisticated approach is needed than aggregating predictions of single scenes by the predicted class majority.

Furthermore, we investigate the effect of the number of analyzed scenes that was theoretically described in Section III-D. Figure 5 visualizes model performance as a function of the number of analyzed scenes of the pattern-based contribution considering all vehicles. For the initial 1000 scenes, only a subset of the pairs of traffic lights and lanes can be predicted. Thus, the metrics cannot directly be compared but indicate a trend. Accuracy, precision and recall reach saturation with an increasing number of scenes.

3) Rejection Method: The rejection method extracts information from scenes where vehicles cross an intersection while ego detects a traffic light state. 13k scenes can be used for the rejection method when considering only ego trajectories. The reason is that only data points with a distance < 1 meter to the intersection entry are used, where not all traffic lights are in the field of view of the Lyft vehicles. This is primarily not a limitation of our method but of the Lyft sensor set. Hence, the rejection method on

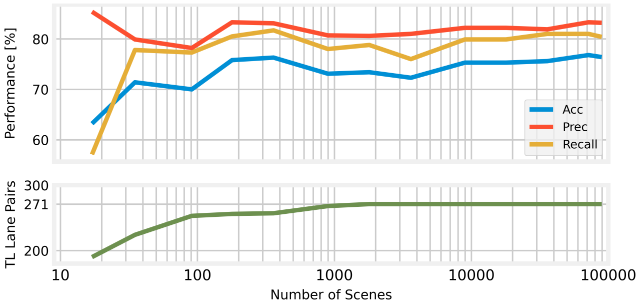

Fig. 5. Visualization of the performance over the number of analyzed scenes of the pattern-based contribution method.

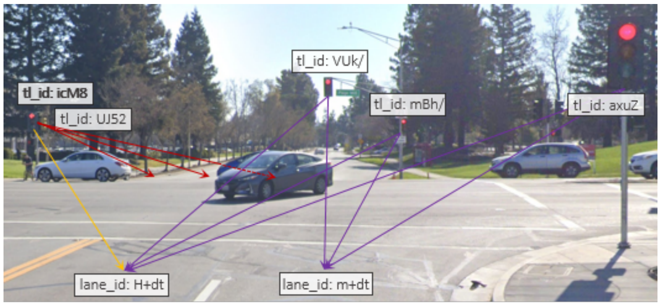

Fig. 6. Lyft dataset error in scene 474 with the true positive (purple), false negative (yellow) and false positive (red) TL2LA labels [23].

ego trajectories reaches a perfect recall of 100 %, but can only classify 55 traffic light and lane pairs.

When using all vehicle trajectories, ego can record other vehicles driving through an intersection from a distance while detecting the traffic light states. This yields 42k usable scenes, and 271 out of 279 traffic light and lane pairs can be classified. The rejection method outperforms the pattern-based contribution method with an F_1 score of 87.2 % and a close to perfect recall of 98.3 %. The precision is 78.3 % because the null hypothesis of a true TL2LA is only rejected with a significant amount of red-light passes. Although precision is lower compared to the pattern-based contribution method, the rejection method is preferable given the importance of a high recall.

C. Qualitative Results

Our approach revealed 20 incorrect TL2LA labels in the Lyft dataset. Assignments are missed in few instances and in scene 474 a dedicated left turn lane is incorrectly assigned to the traffic light controlling the straight direction. The ego vehicle is located on the northwest branch and can detect the state of the traffic light *icM8*, which belongs to the southwest branch. While the semantic map assigns the traffic light *icM8* to the oncoming lanes southwest of the intersection, there must be an assignment to the left-turn lane *H+dt* of the northwest intersection branch. Accordingly, our approach predicts $\mathrm{assign}(icM8, H{+}dt) = 1$, whereas the label defines $\mathrm{assign}(icM8, H{+}dt) = 0$ This is a false negative error in the label. False negative errors are critical because a relevant traffic light is not obeyed. In this example, the vehicle would perform a left turn even if traffic light *icM8* was red. Since the geometry of the traffic light *UJ52* cannot be differentiated in the two-dimensional space from the geometry of *icM8*, it seems obvious that the wrong traffic light was selected during the manual labeling process of the semantic map. This shows that human annotations are error-prone even for small coverages. We manually corrected the false labels and performed all experiments on the corrected dataset.

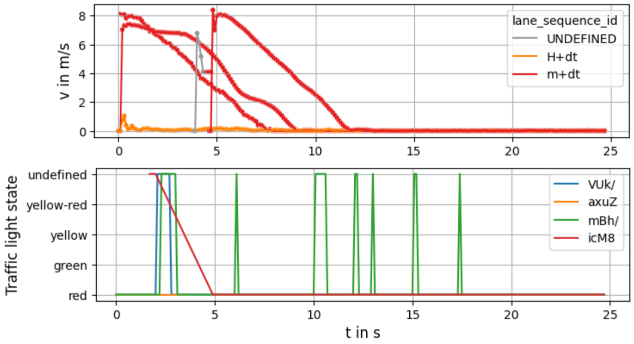

Fig. 7. Graph of vehicle velocities and traffic light states over the time for scene 474. Based on motion patterns and detected traffic light states, the pattern-based contribution method assumes TL2LAs between the lanes *m+dt*, *H+dt* and their traffic lights *VUk/, axuZ, mBh/, icM8*.

Figure 6 shows the real-world perspective (northwest branch) of the ego vehicle in scene 474. Additionally, the TL2LAs are visualized. Yellow shows the assignment of traffic light *icM8* to lane *H+dt*, which is not labeled in the semantic map (false negatives), and red indicates erroneously labeled TL2LAs (false positives).

Figure 7 shows the velocity-time and the traffic light state-time diagrams of scene 474. The colors in the velocity-time

TABLE IV
QUANTITATIVE RESULTS

Method	Scope	Scenes	Vehicles	TL-Lane Pairs	Acc[%]	Prec[%]	Recall[%]	F_1[%]
Naive Baseline	ego only all vehicles	109k 109k	109k 109k	55 271	81.8 66.4	81.8 66.4	**100** **100**	90.0 79.8
Pattern-based Contribution	ego only all vehicles	64k 90k	64k 10M	55 271	83.6 76.4	84.6 **83.2**	97.8 80.4	90.7 81.8
Rejection	ego only all vehicles	13k 42k	13k 124k	55 271	**85.5** **80.8**	**84.9** 78.3	**100** 98.3	**91.8** **87.2**

diagram represent the assignment of the detected vehicles to the two lane sequences in front of the intersection. Grey indicates that the position of a vehicle could not be mapped to a lane sequence. The state-time diagram visualizes the state of the detected traffic lights. Due to occlusion, the traffic light states are noisy and cannot be detected consistently.

VI. Limitations

Our approach to deriving the TL2LA from motion data has certain limitations described in the following.

1) Availability of Traffic Light States and Motion Patterns: A TL2LA is only derived between lanes that vehicles have been recorded on and traffic lights whose state has been detected. We assume that a sensor set is chosen that can detect all relevant traffic lights. Eventually, pairs of traffic lights and lanes that are not covered in the dataset can be conservatively assumed to have a TL2LA. This way, all detected traffic lights are considered, and the safety-critical case of disregarding a true TL2LA is avoided.

2) Disambiguation of synchronized Traffic Lights: No disambiguation is possible if two traffic lights are only recorded in the same state, even if their lane assignments differ. This can be resolved by combining the motion data-based approach with a geometry-based and inlay-based approach.

3) Traffic Light-independent Rules: There are other corner cases such as flashing red lights, which are treated as an all-way stop, vehicles running a red light, or police controlling traffic. Examples are scenes 495, 597, and 1719 in the Lyft dataset, where all traffic lights for the straight lanes are detected as red, but vehicles on all lanes pass the intersection. The applied methods need to be robust to handle such outliers.

VII. Conclusion

In this paper, we presented a novel solution to the problem of learning traffic light to lane assignments by using motion data. Both a pattern-based contribution and a rejection method were implemented and validated to show the trade-off between precision and recall. We found that the rejection method is very effective regarding safety considerations. For future work, a more sophisticated approach could use a generic graph encoding [24] and formulate the TL2LA task as link prediction in graphs.

Using motion patterns and traffic light states, our proposed approach derives the TL2LA independent of the geometric constellation. In deployment, combining this motion-based approach with a geometric approach can yield the best results by either using both sources of information in an integrated model or by having redundancy. Additionally, we proposed a dataset transformation to enable the use of available motion prediction datasets for this task. By providing an API for the Lyft Level 5 dataset, we encourage the research community to invent robust approaches that meet the desired safety requirements.

References

[1] S. Ulbrich, T. Nothdurft, M. Maurer, and P. Hecker, “Graph-based context representation, environment modeling and information aggregation for automated driving,” in *IV*, 2014, pp. 541–547.

[2] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes, D. Ramanan, P. Carr, and J. Hays, “Argoverse 2: Next generation datasets for self-driving perception and forecasting,” in *NeurIPS*, 2021.

[3] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, “nuscenes: A multimodal dataset for autonomous driving,” in *CVPR*, June 2020.

[4] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou, Z. Yang, A. Chouard, P. Sun, J. Ngiam, V. Vasudevan, A. McCauley, J. Shlens, and D. Anguelov, “Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset,” in *ICCV*, October 2021, pp. 9710–9719.

[5] *One Thousand and One Hours: Self-driving Motion Prediction Dataset*, ser. Proceedings of Machine Learning Research, vol. 155. PMLR, 2020.

[6] N. Fairfield and C. Urmson, “Traffic light mapping and detection,” in *2011 IEEE Int. Conf. on Robotics and Automat.*, 2011, pp. 5421–5426.

[7] F. Poggenhans, “Generierung hochdetaillierter karten für das automatisierte fahren,” Ph.D. dissertation, 2019.

[8] Q. Li, Y. Wang, Y. Wang, and H. Zhao, “Hdmapnet: An online hd map construction and evaluation framework,” 2022.

[9] Y. Zhou, Y. Takeda, M. Tomizuka, and W. Zhan, “Automatic construction of lane-level hd maps for urban scenes,” in *IROS*, 2021, pp. 6649–6656.

[10] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, “Bevformer: Learning bird’s-eye-view representation from multi-camera images via spatiotemporal transformers,” in *ECCV*. Springer Nature Switzerland, 2022, pp. 1–18.

[11] M. Elhousni, Y. Lyu, Z. Zhang, and X. Huang, “Automatic building and labeling of hd maps with deep learning,” *AAAI*, pp. 13 255–13 260, 2020.

[12] T. Langenberg and F. Wörgötter, “Automatic traffic light to ego vehicle lane association at complex intersections,” *ITSC*, vol. 2018-Novem, pp. 1350–1357, 2018.

[13] G. Yuan, P. Sun, J. Zhao, D. Li, and C. Wang, “A review of moving object trajectory clustering algorithms,” *Artif. Intell. Rev.*, vol. 47, pp. 123–144, Jan. 2017.

[14] L. Jiao, Y.-L. Wu, G. Wu, E. Chang, and Y. Wang, “Anatomy of a multicamera video surveillance system,” *Multimed. Syst.*, vol. 10, pp. 144–163, 08 2004.

[15] Y. Chen and J. Krumm, “Probabilistic modeling of traffic lanes from gps traces,” in *ACM*, Nov. 2010, pp. 81–88.

[16] E. Uduwaragoda, A. Perera, and S. Dias, “Generating lane level road data from vehicle trajectories using kernel density estimation,” in *ITSC*, 2013, pp. 384–391.

[17] C. Doer, M. Henzler, H. Messner, and G. F. Trommer, “Hd map generation from vehicle fleet data for highly automated driving on highways,” in *IV*, 2020, pp. 2014–2020.

[18] M. Liebner, D. Jain, J. Schauseil, D. Pannen, and A. Hackelöer, “Crowdsourced hd map patches based on road model inference and graph-based slam,” in *IV*, 2019, pp. 1211–1218.

[19] J. Shu, S. Wang, X. Jia, W. Zhang, R. Xie, and H. Huang, “Efficient lane-level map building via vehicle-based crowdsourcing,” *ITSC*, 2020.

[20] A. Derrow-Pinion, J. She, D. Wong, O. Lange, T. Hester, L. Perez, M. Nunkesser, S. Lee, X. Guo, B. Wiltshire *et al.*, “ETA prediction with graph neural networks in google maps,” in *Proceedings of the 30th ACM Int. Conf. on Information & Knowledge Management*, 2021, pp. 3767–3776.

[21] F. Wirthmüller, J. Hipp, C. Reichenbächer, and M. Reichert, “The atlas of lane changes: Investigating location-dependent lane change behaviors using measurement data from a customer fleet,” *ITSC*, pp. 1225–1232, 2021.

[22] C.-k. Wong and Y.-y. Lee, “Lane-based traffic signal simulation and optimization for preventing overflow,” *Mathematics*, vol. 8, 2020.

[23] Google, “Street view, 2798 hanover st, palo alto, california,” February 2020. [Online]. Available: https://goo.gl/maps/HGbaZ7Bk512tUmJm7

[24] T. Monninger, J. Schmidt, J. Rupprecht, D. Raba, J. Jordan, D. Frank, S. Staab, and K. Dietmayer, “Scene: Reasoning about traffic scenes using heterogeneous graph neural networks,” *IEEE Robot. Autom. Lett.*, 2023.