Constellation Codebooks for Reliable Vehicle Localization

Isabell Hofstetter\textsuperscript{(E)}, Malte Springer, Florian Ries, and Martin Haueis

Mercedes-Benz AG, Sindelfingen, Germany
\{isabell.hofstetter,malte.springer,florian.ries,martin.haueis\}@daimler.com

Abstract. Safe feature-based vehicle localization requires correct and reliable association between detected and mapped localization landmarks. Incorrect feature associations result in faulty position estimates and risk integrity of vehicle localization. Depending on the number and kind of available localization landmarks, there is only a limited guarantee for correct data association due to various ambiguities.

In this work, a new data association approach is introduced for feature-based vehicle localization which relies on the extraction and use of unique geometric patterns of localization features. In a preprocessing step, the map is searched for unique patterns that are formed by localization landmarks. These are stored in a so called codebook, which is then used online for data association. By predetermining constellations that are unique in a given map section, an online guarantee for reliable data association can be given under certain assumptions on sensor faults.

The approach is demonstrated on a map containing cylindrical objects which were extracted from LiDAR data. The evaluation of a localization drive of about 10 min using various codebooks both demonstrates the feasibility as well as limitations of the approach.

Keywords: Vehicle localization · Geometric hashing · Data association

1 Introduction

High definition (HD) maps are essential to enable safe highly automated driving. Consequently, reliable and accurate vehicle localization in such maps is inevitable. In the past years, various feature-based localization approaches have been introduced. These methods utilize a separate map layer containing localization features, also called landmarks, which they aim to associate with detected features in the vehicle’s surrounding. The resulting association constraints pose an optimization problem, which is solved for an accurate pose estimate.

State-of-the-art methods differ mainly in the kind of localization features that are used. Whereas some approaches rely on dense point clouds such as visual point features \cite{17} or RADAR point clouds \cite{16}, other methods make use of landmarks such as road markings \cite{11,15}, or geometric primitives \cite{4,7}.
However, all these methods imply some kind of feature association. Especially in those cases where localization features are sparse and do not have an expressive descriptor, data association represents one of the main challenges for reliable vehicle localization. To satisfy integrity requirements, a guarantee for correct data association is desirable.

This contribution focuses on the extraction and use of unique landmark patterns, hereafter referred to as constellation codebooks, for reliable data association. Landmarks used for localization often appear in periodic orders, so that many ambiguities arise in the data association step. This contribution suggests to identify that specific geometric information which breaks the symmetry and to cluster landmarks such that the resulting feature patterns reflect the uniqueness of the environment. This enables reliable feature association even in sparse feature maps.

The proposed approach consists of a two-step process: In a first step, a given map of localization features is searched for such unique geometric patterns of landmarks. In this work, these constellations are referred to as codewords. All extracted codewords are stored in a codebook, which can then be used online for vehicle localization. For easy and fast search and access, this work suggests to store the codebook in a hash table format.

During the localization drive, the geometric patterns of detected localization features are compared to the codebook content. As soon as one of the codewords is detected online, incorrect associations can be excluded and a certain guarantee can be given that the resulting associations are correct. Hereby, the suggested data association approach, which is based on Geometric Hashing [19], does not make any assumptions on prior vehicle poses. It allows for feature association based only on the geometric information provided by the extracted features, while ambiguities are eliminated due to the preprocessing step.

The method is derived and tested on a dataset representing an urban loop of 3.8 km in Sindelfingen, Germany, where pole-like features are used as localization landmarks. The method, however, is not limited to this specific kind of features and can easily be adapted to arbitrary localization features. Experimental results verify the potential of the proposed approach.

2 Related Work

In order to provide reliable localization that meets high integrity requirements, most approaches rely on outlier detection and exclusion [1,21] or pattern matching techniques [2,7] to avoid incorrect associations between map and sensor data. By accumulating sensor features and matching a group of landmarks with the map, these approaches try to reduce or even eliminate the number of ambiguities in data association and exclude associations that are found to be outliers. A major drawback of most approaches is the assumption that a certain number of correct associations is found and only a small percentage of outliers needs to be eliminated. Also, a guarantee that all outliers are excluded or that the correct alignment with the map is found cannot be given.
Brenner et al. suggest a pattern matching algorithm in [4,14] for vehicle localization. By deriving a descriptor for cylindrical localization features based on other features in the near surrounding, they notice that such patterns can be ambiguous as well and not every accumulated pattern of localization landmarks provides enough unique information for a reliable alignment with the map. The existence of ambiguities in data association is also discussed in [6], where a method based on Geometric Hashing [8] for the extraction of such patterns is suggested. Applying the same Geometric Hashing algorithm for the actual feature association step has been introduced in [5] and a similar technique is used in this work. The main contribution of this work, however, focuses on the generation and use of constellation codebooks, i.e. unique geometric patterns of localization features.

Extracting unique information from the environment for localization has been considered in works like [9]. Here, a method is proposed for creating unique identifiers, called fingerprint sequences, for visually distinct locations using panoramic color images. Also, object recognition methods like Bag of Words approaches as utilized in [10,20] are well known and provide a similar kind of vocabulary, however, without satisfying any uniqueness requirements. Furthermore, these methods have only been applied in the field of image data processing. To the authors’ knowledge, the concept of identifying unique patterns in feature maps and the use of codebooks for localization has not been studied before.

3 Codebook Generation

In the following, a codebook is defined as a set of feature patterns, hereafter referred to as codewords. The codebook is computed for a given map or map section and used for feature-based vehicle localization. The proposed feature constellations, which are contained in a codebook must satisfy certain constraints in order to enable reliable vehicle localization. The main requirement is the uniqueness of the extracted constellations in a given map section. If a feature pattern is known to be unique in a map even under various sensor or map faults, this information can be used for reliable data association.

In the following subsections, the codebook generation is described in detail. This involves the derivation of various requirements for the constellations of interest, as well as a search algorithm for the identification of such feature patterns.

3.1 Definition of Uniqueness

Feature patterns of interest are desired to be unique in a sense that they are uniquely associable in a given map section. This must still be true, even if the measurements of the landmarks are affected by noise. In other words: For a unique feature constellation, there exists no other approximative matching in the map section of interest than the correct one.
To describe these assumptions in a mathematical way, the distance $d$ between single landmarks $p = (x, y) \in \mathbb{R}^2$ and sets of landmarks $G = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, $(x_i, y_i) \in \mathbb{R}^2$, $i = 1, \ldots, N$, is defined as

$$d(p, G) := \min\{ ||p - g||_2 \mid g \in G \},$$

which is the distance of $p$ to its nearest neighbor in $G$. Thus, the mean point-to-point distance between two sets of landmarks $G_1$ and $G_2$ with cardinality $|G_1| = |G_2|$ can be defined as

$$D(G_1, G_2) := \frac{1}{|G_1|} \sum_{g \in G_1} d(g, G_2).$$

Let $T$ be the set of all euclidean transformations, i.e. all combinations of rotation and translation. Then, a pattern of landmarks $C$ is unique in a map section $M$ if

$$\forall T_{eucl} \in T \nexists F \subset M \text{ with } |F| = |C| : D(T_{eucl}(C), F) < \epsilon$$

for $F \neq C$ and a minimum distance $\epsilon$.

This means that considering any arbitrary transformation $T_{eucl} \in T$, no subset $F \subset M$ with $|F| = |C|$ exists, whose mean distance to $T_{eucl}(C)$ is smaller than $\epsilon$. Therefore, there exists only one transformation, which allows an approximative matching [3] of the constellation with the map, which is the identity transformation.

In the following, $\epsilon$ is also called uniqueness threshold, because the choice of $\epsilon$ is decisive for the classification of feature patterns in unique or ambiguous patterns. For large $\epsilon$, less patterns will be identified as unique. On the other hand, for small $\epsilon$, patterns that differ only slightly in their geometrical characteristics will be classified as unique, which results in less ambiguous patterns.

### 3.2 Codeword Extraction

In practice, it is not feasible to find unique feature patterns in large maps of entire cities or even countries. However, a reasonable assumption is that an initial, very rough position estimate within a map is given. Therefore, the patterns of interest do not need to be unique within an entire map, but rather in a map section of limited size. In the following, the uniqueness requirement (3) will be limited to certain map tiles $T$ of size $R$. Thus, $R = \infty$ refers to the entire map.

Also, to facilitate the detection of codewords $C$ during the localization drive, the spatial expansion of codewords, i.e. the diameter of the smallest enclosing circle containing a codeword, should be limited. This can be described as

$$d(p_i, p_j) \leq L \ \forall \ p_i, p_j \in C, \ i \neq j,$$

with a given maximum diameter $L$. 

Algorithm 1. Codebook Generation

**Input:** Map $M$, tile radius $R$, max. expansion $L$, uniqueness threshold $\epsilon$.

**Output:** Codebook $CB$ containing all unique constellations with cardinality $s_{cw}$.

1. Generate overlapping map tiles $\{T_1, \ldots, T_n\}$ of size $R$ covering the whole map $M$.
2. Gather map features for each tile $T_i = \{p_1, \ldots, p_{|T_i|}\}$, $i = 1, \ldots, n$.
3. For each tile $T_i$, construct all possible feature combinations $C$ of size $s_{cw}$ fulfilling expansion requirement (4) for the given maximum expansion $L$.
4. Check uniqueness requirement (3) with given uniqueness threshold $\epsilon$ for every constellation $C$ regarding the current map tile.
5. If $C$ fulfills (3), add it to the codebook $CB$. Discard $C$ otherwise.

Now, in order to generate a constellation codebook, landmark patterns with characteristics (3) and (4) can be identified in a map $M$. The method is summarized in Algorithm 1.

In steps 1) and 2), the map is divided into map tiles of a given radius $R$. This parameter should be chosen in a way that the current map tile can always be found reliably, for example using a low-cost GPS module. Then, each tile is regarded separately.

First, all possible feature combinations of the desired cardinality $s_{cw}$ and expansion $L$ are constructed. These are the candidate patterns which need to be classified into unique or ambiguous patterns.

Regarding a tile $T_i$ containing $N_f$ features, there are a total of $\binom{N_f}{s_{cw}}$ constellations of $s_{cw}$ features. As an example, for $N_f = 50$ and $s_{cw} = 3$, there exist $\binom{50}{3} = 19,600$ constellations in total. For larger patterns of cardinality $s_{cw} = 6$, there are almost 16 million different patterns. However, by limiting the constellation expansion to $L$ as suggested by Eq. (4), this number of candidate patterns can be reduced noticeably.

All candidate patterns found in step 3) are then compared to each other and uniqueness requirement (3) is checked for a given uniqueness threshold $\epsilon$. Here, scan matching methods like the well-known Iterative Closest Point (ICP) algorithm [13] can be used. If the constellation proves to fulfill the uniqueness requirement, it is added to the codebook. If it is not unique according to (3), the pattern is discarded. Finally, steps 3) – 5) are repeated for each previously constructed map tile $T_i$.

Figure 1 shows example constellations for the first map tile $T_0$ with tile radius $R = 200$ m. Map features are plotted in gray. Codewords are visualized by colored lines connecting the features belonging to the codeword.

4 Localization Using Constellation Codebooks

The offline generated constellation codebook can now be used for the purpose of data association for vehicle localization. In order to allow an easy and fast recognition of codewords and a quick search for associations, in this work, a Geometric
Hashing based method is proposed for the representation of codebooks as well as the actual data association step. Geometric Hashing was initially introduced in [8] for object recognition purposes. It is a popular technique for the association of geometric features that have undergone transformations. The method consists of a training phase as well as the recognition phase. During the training phase, features of interest are represented in a variety of coordinate systems that are defined by the features themselves and stored in a quickly searchable, tabular format. Online, a given input sample $S$ can be searched for the objects of interest with the help of the previously generated hash table.

The following subsections will discuss a modified hashing algorithm which is suggested in combination with constellation codebooks as they were derived in Sect. 3. For further details on Geometric Hashing, the authors refer to [8,19] or [12].

### 4.1 Hash Table Generation

For the generation of the hash table, which is used for codeword recognition during localization, each codeword is hashed separately. Details on the hashing procedure as well as the noise model will be discussed briefly.

**Hashing of Codewords.** Let $CB$ be a codebook that contains a number of codewords $CB = \{c_1, c_2, \ldots, c_N\}$. Each codeword $c_i$ consists of $s_{cw}$ features $c_i = \{p_{i1}^1, \ldots, p_{i s_{cw}}^i\}$, $i = 1, \ldots, N$. Each ordered pair of features $(p_{in}^i, p_{im}^i)$ with $p_{in}^i, p_{im}^i \in c_i$ and $n \neq m$ represents a *basis pair*. Let $K_i$ be the set of basis pairs of codeword $c_i$. The cardinality of this basis set is $|K_i| = s_{cw}(s_{cw} - 1)$. Each basis $k = (p_{in}^i, p_{im}^i) \in K_i$ defines a local coordinate frame $O_{(p_{in}^i,p_{im}^i)}$, called the *geometric*
basis. For each basis $k \in K$, the remaining features $c_i \setminus k$ are transformed into the corresponding basis frame and quantized according to a grid. Finally, a hash function is applied to each feature location and the hashes are stored in a hash table $H$. The main parameters used in the hashing procedure are visualized in Fig. 2.

**Noise Model.** If the measured features are affected by noise, the positional error in the hash locations has to be described by a valid noise model. Obviously, the positional noise on the feature locations in the hash frame depends heavily on the chosen basis pair as well as the position of the features within the geometric basis. A small distance between basis pair features will enhance the noise in angular direction, while the noise in radial direction stays constant. Therefore, the authors suggest to estimate the resulting noise distribution in polar coordinates. For the underlying sensor noise, a Gaussian distribution with standard deviation $\sigma$ is assumed, which is centered at the “true” location of the feature. A Monte Carlo Simulation is performed to generate noisy measurements, which are then transformed into the basis frame. For each geometric basis and each feature within that basis, the standard deviation in angular $\sigma_\theta$ and radial $\sigma_r$ component are estimated using these simulated measurements and stored in the hash table for the online use.

![Fig. 2.](image)

**Fig. 2.** Visualization of the Geometric Hashing procedure. An example codeword of size $s_{cw} = 5$ (green) is hashed according to the basis frame $O_{(x_{hash}, y_{hash})}$ defined by the features $p_1$ and $p_2$. Other map features not belonging to the codeword (gray) are discarded in the hashing procedure. (Color figure online)

### 4.2 Feature Association

Given a sample of detected features $D$ and the previously generated hash table $H$, feature associations can now be determined by searching the hash table. For more detail on data association for vehicle localization based on Geometric
Hashing the authors refer to [5]. In the following, the method will be briefly recapitulated.

First, two arbitrary detections $p_1, p_2 \in D$ are chosen, which define a geometric basis $O(p_1, p_2)$, just as described before. All feature detections $D$ are then transformed into this basis by a rigid transformation, quantized, and hashed accordingly. For each computed hash value, appropriate bins in the hash table $H$ are accessed and a vote is given to each basis found there. In this step, not only the exact bin is accessed but also its direct neighborhood to ensure that noisy measurements that fell into neighboring bins can be associated as well.

Then, the set of bases $B$ is determined that received as many votes as the number of features belonging to the codeword, namely $s_{cw}$. These bases represent those codewords that could potentially be matched. Finally, the resulting candidate associations are validated based on the Mahalanobis distance between feature measurement in the hash frame and noise distribution, which is stored in the hash table.

If valid associations were found, the matching codeword is returned. Otherwise, the previously described steps can be repeated for each available basis pair $k = (p_i, p_j)$ with $p_i, p_j \in D, i \neq j$. In practice, not every generated basis will necessarily have a matching basis in the hash table. If a basis pair was chosen, which is not part of a codeword, the resulting basis cannot be matched with the hash table. Therefore, multiple bases have to be generated in order to find one that is part of a codeword.

5 Experimental Results

5.1 Dataset

The vehicle BerthaONE [18] was used for the acquisition of the dataset utilized to demonstrate and evaluate the proposed approach. A loop of about 3.8 km in an urban area in Sindelfingen, Germany, which results in about 10 min of driving time, was used for the evaluation. The corresponding localization map layer contains 593 cylinders such as traffic signs, tree trunks, and traffic lights as localization features, which were extracted from LiDAR data as described in [7]. Each cylinder is described by its center point $(x, y) \in \mathbb{R}^2$ on the ground plane. The feature locations in map coordinates were shown in Fig. 1. A reference solution for validation is provided by [17].

5.2 Localization Framework

The localization framework used to demonstrate the approach is visualized in Fig. 3. A State-of-the-Art localization framework consisting of map, feature detectors, feature association, and optimizer is adapted and extended by a codebook generator and hashing functionalities. The main contribution of this work, namely the codebook generation and feature association, is highlighted in green. The hashing functionalities are highlighted in gray.
Additional sensors, such as a GNSS receiver and an odometry unit, are added to the framework in dashed lines. These are optional components that are not needed for the data association step itself. However, the GNSS input can be used to determine the current map tile and odometry measurements could be utilized to improve the continuity of the localization solution whenever the suggested method is unavailable.

However, one of the biggest advantages of the proposed approach lies in the fact that no prior pose propagation is needed to find correct associations. Therefore, such information is completely neglected in this work.

5.3 Constellation Codebooks

To evaluate the described approach, a variety of codebooks is generated for the before mentioned map and used for localization. Parameters for the codebook generation that were examined in this work are the tile size $R$, the uniqueness threshold $\epsilon$ as well as the number of resulting codewords. All generated codebooks contain feature constellations of size $s_{cw} \in \{3, 4\}$ with a maximum constellation expansion of $L = 35\,\text{m}$, which corresponds to the sensor range. Constellations of size $s_{cw} > 4$ could also be extracted using the same method. However, this is something that was not evaluated, since no enhancement of the system availability is expected. Also, it is desirable to extract the minimal number of landmarks required to provide a reliable localization solution.

Depending on the chosen tile radius $R$ and the uniqueness threshold $\epsilon$, a different number of feature patterns was identified as unique patterns in the map ranging in between 0 up to 550,000 codewords. The numbers can be taken from Fig. 4. Here, codebooks were generated for tiles of radius $R \in \{\infty, 300\,\text{m}, 200\,\text{m}\}$ and uniqueness threshold $\epsilon \in \{0.25\,\text{m}, 0.5\,\text{m}, 0.75\,\text{m}, 1.0\,\text{m}, 1.25\,\text{m}\}$. For large uniqueness thresholds, the number of unique feature patterns decreases as
expected. While there are more than 400.000 codewords using a tile size of $R = 300\text{ m}$ and $\epsilon = 0.25\text{ m}$, only 129 codewords are available for $\epsilon = 1.25\text{ m}$ and the same tile size. In the case of $R = \infty$, no codewords were found at all for $\epsilon > 0.75\text{ m}$. Also, for decreasing tile sizes, the number of codewords increases. Looking at $\epsilon = 0.5\text{ m}$, there are 17.802 codewords if the uniqueness is required in the whole map. For $R = 300\text{ m}$, this number increases to 62.939, and for $R = 200\text{ m}$ to 104.491. As an example, a codebook for $R = \infty$ and $\epsilon = 0.5\text{ m}$ containing 17.802 codewords of 3 or 4 features is visualized in Fig. 5. Each color represents the features that are part of a different codeword. It should also be noted that there are some parts in the map where the codeword density is relatively low. This is due to the low feature density in those regions.

![Number of codewords vs Uniqueness threshold](image)

**Fig. 4.** Number of extracted codewords of size $s_{\text{cw}} \in \{3, 4\}$ for various tile sizes $R$ and uniqueness thresholds $\epsilon$.

![Example codebook](image)

**Fig. 5.** Example codebook with feature patterns of size $s_{\text{cw}} \in \{3, 4\}$ for tile radius $R = \infty$ and uniqueness threshold $\epsilon = 0.5\text{ m}$. Each color represents one codeword. Features that belong to one codeword are connected by lines.

### 5.4 Localization Results

The potential of the proposed approach is mainly assessed by the availability of the system and the percentage of correct associations. The association
availability obviously depends very strongly on the codebook used and especially on the number of codewords contained in the codebook. This correlation can be observed in Fig. 6. The use of codebooks that contain a few hundred codewords results in a low availability of less than 5%. For larger codebooks with more than 500,000 codewords, an association availability of about 48% is reached. This is due to the fact that map features often are occluded or not yet in sensor range. Only if all features of one codeword are detected, the association is possible and the system is available. Therefore, the system availability increases with the number of codewords that are contained in the codebook used. The percentage of correct associations is visualized in Fig. 7. Here, the uniqueness threshold $\epsilon$, which was chosen to extract the codewords, plays an important role. For a relatively small choice of $\epsilon = 0.25$ m, codewords were correctly associated in up to 97% of the time, depending on the tile size $R$. By increasing the uniqueness threshold up to 1 m, 100% correct associations are found independent of the tile size $R \in \{200$ m, 300 m$\}$. For $R = \infty$ and $\epsilon > 0.75$ m no codewords are available, therefore the system is not available.

Overall, there exists a trade-off between system availability and reliability. In order to reach a high availability, it is desirable to extract many codewords. This can be achieved using small tiles and small uniqueness thresholds. On the other hand, for reliable associations, a high uniqueness threshold $\epsilon$ should be chosen which results in fewer codewords, but 100% correct associations. In practice, it
is conceivable to use multiple codebooks with different uniqueness thresholds to boost availability, where no high integrity solution is available.

6 Conclusions

In this work, a new approach for reliable data association for feature-based vehicle localization was proposed. The method suggests to preprocess a given map in order to identify and extract unique geometric landmark patterns, called *code-words*, which are suitable for reliable feature association. These unique feature constellations are hashed and stored in a hash table format, the *constellation codebook*, for quick and easy search and access during localization.

The exclusive use of unique landmark patterns for data association enables, under certain assumptions on sensor faults, a guarantee for correct feature association and, therefore, provides reliable vehicle localization.

Experimental results were discussed for a feature map of about 3.8 km of roads containing about 600 cylindrical localization landmarks. A variety of codebooks with different parameter choices was constructed for this map. The evaluation of a localization drive of about 10 min in the before mentioned map using these previously generated codebooks demonstrates a trade-off between system availability and reliability.

Future work will focus on the use of multiple constellation codebooks that satisfy different uniqueness requirements for a continuous and accurate localization solution.

References

1. Alsayed, Z., Bresson, G., Verroust-Blondet, A., Nashashibi, F.: Failure detection for laser-based SLAM in urban and peri-urban environments. In: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), pp. 1–7 (2017)

2. Ascani, A., Frontoni, E., Mancini, A., Zingaretti, P.: Feature group matching for appearance-based localization. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3933–3938 (2008)

3. Bishnu, A., Das, S., Nandy, S.C., Bhattacharya, B.B.: A simple algorithm for approximate partial point set pattern matching under rigid motion. In: Rahman, M.S., Fujita, S. (eds.) WALCOM 2010. LNCS, vol. 5942, pp. 102–112. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-11440-3_10

4. Brenner, C.: Global localization of vehicles using local pole patterns. In: Denzler, J., Notni, G., Süße, H. (eds.) DAGM 2009. LNCS, vol. 5748, pp. 61–70. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03798-6_7

5. Hofstetter, I., Sprunk, M., Ries, F., Haueis, M.: Reliable data association for feature-based vehicle localization using geometric hashing methods. In: 2020 IEEE International Conference of Robotics and Automation (ICRA). IEEE (2020)

6. Hofstetter, I., Sprunk, M., Schuster, F., Ries, F., Haueis, M.: On ambiguities in feature-based vehicle localization and their a priori detection in maps. In: 2019 IEEE Intelligent Vehicles Symposium (IV), pp. 1192–1198. IEEE (2019)
7. Kümmerele, J., Sons, M., Poggenhans, F., Lauer, M., Stiller, C.: Accurate and efficient self-localization on roads using basic geometric primitives. In: 2019 International Conference on Robotics and Automation (ICRA), pp. 5965–5971. IEEE (2019)
8. Lamdan, Y., Wolfson, H.J.: Geometric hashing: A general and efficient model-based recognition scheme (1988)
9. Lamon, P., Nourbakhsh, I., Jensen, B., Siegwart, R.: Deriving and matching image fingerprint sequences for mobile robot localization. In: 2020 IEEE International Conference of Robotics and Automation (ICRA), vol. 2, pp. 1609–1614. IEEE (2001)
10. Lavoué, G.: Combination of bag-of-words descriptors for robust partial shape retrieval. Vis. Comput. 28(9), 931–942 (2012)
11. Poggenhans, F., Salscheider, N.O., Stiller, C.: Precise localization in high-definition road maps for urban regions. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2167–2174 (2018)
12. Rigoutsos, I.: Massively Parallel Bayesian Object Recognition. New York University, Department of Computer Science, Technical Report (1992)
13. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Proceedings third international conference on 3-D digital imaging and modeling, pp. 145–152. IEEE (2001)
14. Schlichting, A., Brenner, C.: Localization using automotive laser scanners and local pattern matching. In: 2014 IEEE Intelligent Vehicles Symposium Proceedings, pp. 414–419. IEEE (2014)
15. Schreiber, M., Knöppel, C., Franke, U.: Laneloc: Lane marking based localization using highly accurate maps. In: IEEE Intelligent Vehicles Symposium (IV), pp. 449–454 (2013)
16. Schuster, F., Keller, C.G., Rapp, M., Haueis, M., Curio, C.: Landmark based radar SLAM using graph optimization. In: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), pp. 2559–2564 (2016)
17. Sons, M., Lauer, M., Keller, C.G., Stiller, C.: Mapping and localization using surround view. In: IEEE Intelligent Vehicles Symposium (IV), pp. 1158–1163 (2017)
18. Taş, Ö.S., Salscheider, N.O., et al.: Making bertha cooperate-team annieways entry to the 2016 grand cooperative driving challenge. IEEE Trans. Intell. Transp. Syst. 19(4), 1262–1276 (2017)
19. Wolfson, H.J., Rigoutsos, I.: Geometric hashing: An overview. IEEE Comput. Sci. Eng. 4(4), 10–21 (1997)
20. Yuan, J., Wu, Y., Yang, M.: Discovery of collocation patterns: from visual words to visual phrases. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)
21. Zinoune, C., Bonnifait, P., Ibañez-Guzmán, J.: Sequential FDIA for autonomous integrity monitoring of navigation maps on board vehicles. IEEE Trans. Intell. Transp. Syst. 17(1), 143–155 (2016)