Sequence-based mapping for probabilistic visual loop-closure detection

Konstantinos A. Tsintotas[1], Loukas Bampis[1], Shan An[2], George F. Fragulis[3], Spyridon G. Mouroutsos[4], and Antonios Gasteratos[1]

2021 IEEE International Conference on Imaging Systems and Techniques (IST) | 978-1-7281-7371-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/IST50367.2021.9651458

***Abstract*— During simultaneous localization and mapping, the robot should build a map of its surroundings and simultaneously estimate its pose in the generated map. However, a fundamental task is to detect loops, i.e., previously visited areas, allowing consistent map generation. Moreover, within long-term mapping, every autonomous system needs to address its scalability in terms of storage requirements and database search. In this paper, we present a low-complexity sequence-based visual loop-closure detection pipeline. Our system dynamically segments the traversed route through a feature matching technique in order to define sub-maps. In addition, visual words are generated incrementally for the corresponding sub-maps representation. Comparisons among these sequences-of-images are performed thanks to probabilistic scores originated from a voting scheme. When a candidate sub-map is indicated, global descriptors are utilized for image-to-image pairing. Our evaluation took place on several publicly-available datasets exhibiting the system's low complexity and high recall compared to other state-of-the-art approaches.**

I. Introduction and related work

Simultaneous localization and mapping (SLAM) [1], i.e., a robot's ability to create a map of its surroundings and then estimate its position in it, has evolved over the last three decades as the most precious asset for autonomous navigation [2]. However, due to the sensor's noise and the absence of global positioning measurements, the robot's estimation concerning its pose (position and orientation) is prone to drift over time. Hence, a crucial task that every SLAM system has to address is the identification of previously visited map regions in order to bound and rectify the accumulated drift. The detection of such events, also known as loop-closures, facilitates the SLAM's consistent map generation.

In the early years, several methods were exploited to map the robot's environment based on range and bearing sensors [3]. Still, in recent years, the increased processing power in modern computers and the researchers' findings regarding how animals navigate using vision [4] have led to the adoption of cameras as the primary perception unit [5], [6]. Moreover, the low cost and the rich information they provide have made them applicable to several mobile platforms with limited computational capabilities, such as unmanned aerial vehicles (UAVs) [7], [8], [9] and space exploration rovers [10], [11], [12]. In addition, using the visual information of the incoming image stream, loop-closure detection pipelines can perform with high efficiency. Such frameworks includes three main processes: *feature extraction*, *environment representation*, and *decision-making* [13]. The first is related to image processing which extracts visual features for representing the sensory data, the second regards the robot's internal map formulation, while the last is responsible for measuring the system's confidence about its location.

Visual features are divided into global, which are based on the entire image, and local, referring to regions-of-interest. Methods belonging to the former category describe the image's appearance using a single vector [14], [15], [16], [17], [18], [19], [20], while the ones in the second category describe salient local image regions, resulting in a multitude of description vectors [21], [22], [23], [24], [25], [26], [27]. Based on the above, the main advantages of global techniques are their compact representation and their increased computational efficiency during matching. In contrast to the above, local features show robustness against transformations, such as scale and rotation [28]. Aiming to exploit the advantages of both techniques, the robotics community borrowed the model of bag-of-words from text retrieval [29] in order to effectively address the loop-closure detection task. Within this model, each extracted local feature is assigned to a visual word belonging to a visual vocabulary generated beforehand [30]. This way, the incoming image is represented by a visual word histogram [31], [32], [33], [34], [35]. Nevertheless, this line of approaches is highly depended on the quality of their vocabulary and, in turn, their training data [36]. Aiming to improve the performance of a pre-trained vocabulary, incrementally generated visual words are proposed, which are typically produced by clustering similar description vectors along the course of navigation [37], [38], [39], [40], [41], [42], [43], [44].

Regardless of the visual features used for representing the camera data, *environment representation* is distinguished into single- and sequence-based methods. The former approaches use each image as an individual observation to represent the traversed route [45], [46], [47], while the latter generate sub-maps, i.e., groups of individual images, represented by common data [48], [49], [50]. This way, sequence-based frameworks take advantage of the additional information provided by a group of images in a scene [51]. The main difference among the techniques mentioned above regards

[1]Authors are with the Department of Production and Management Engineering, Democritus University of Thrace, 12 Vas. Sophias, GR-671 32, Xanthi, Greece {ktsintot, lbampis, agaster}@pme.duth.gr

[2]Shan An is with the with AR/VR department, JD.com, Beijing, 100191, China, anshan@jd.com

[3]George F. Fragulis is with the Department of Electrical and Computer Engineering, University of Western Macedonia, Kila. GR-50 100, Kozani, Greece, gfragulis@uowm.gr

[4]Spyridon G. Mouroutsos is with the Department of Electrical and Computer Engineering, Democritus University of Thrace, Kimmeria. GR-67 100, Xanthi, Greece, sgmour@ee.duth.gr

their computational complexity while the system searches for loop-closure event. Single-based techniques present higher complexity as the robot has to seek in the whole database at each time-step [52], [53]. In contrast, when sub-maps are adopted, searching is performed among the generated sequences, inspecting, this way, only the most relative locations and images [54], [55]. However, many challenges arise when breaking the map into sequences, including optimal sub-map size, sub-map overlap during query, or consistent semantic map segmentation. To overcome them, several approaches utilize a sliding window of fixed size [56], [57]. Although this solution can improve the system's performance, its functionality can be proved computationally intensive [58].

Finally, *decision-making* refers to the means for obtaining the system's confidence about a loop event. In general, when global features are used, comparisons are made based on their descriptors' distance [59], e.g., Euclidean or cosine [60]. However, for methods based on local features, voting schemes are adopted [61].

As the storage requirements needed to map the whole environment in long-term applications constitute a crucial factor, this paper exploits the advantages mentioned above and proposes a low-complexity system for detecting loop-closures through the scene's appearance. Our pipeline relies on the trajectory's dynamic sub-map definition, while an incremental visual vocabulary is generated for their representation. Also, global descriptors are used for each image included in a sequence permitting faster image-to-image indexing during the query. Our system's confidence about a loop-closure event comes from the probabilistic score originated from the sub-maps' accumulated votes, which are cast during the query. Our main contributions are summarized as follows:

- A straightforward sequence-based visual loop-closure pipeline with reduced computational complexity is proposed.
- An extended experimental evaluation based on the vocabulary's size, the storage requirements, and the operational timings is presented to fully assess the performance of our proposal on publicly available datasets.

The rest of the paper is organized as follows. Section II describes the proposed system in detail. In Section III, its performance is evaluated, while Section IV provides our conclusions and plans for future work are provided.

II. Methodology

A pipeline of three operations is followed to detect loop-closures in the robot's traversed route, namely: i) the image processing, ii) the mapping, and iii) the database query process. Visual features' extraction is associated to the first part, the trajectory segmentation and database creation to the second, whereas the voting scheme and its probabilistic scoring belong to the last part. In the following subsections, each stage is described in detail, while Fig. 1 presents a their schematic representation.

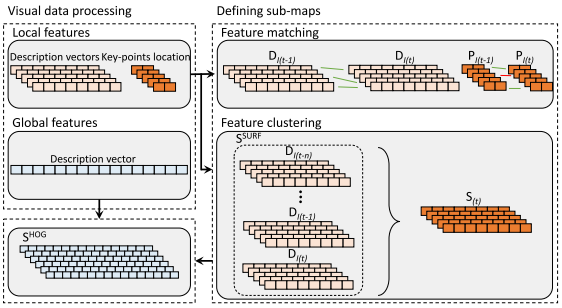

Fig. 1: An overview of the proposed framework. At each time step, local and global visual features are extracted from the obtained image. Firstly, local feature matching is used to dynamically segment the trajectory, and subsequently, the corresponding visual words are generated via growing neural gas clustering [62]. Global features are stored for image-to-image matching when the proper sequence is detected.

A. Visual data processing

As the visual stream is captured by the camera sensor, the image processing module is responsible for two processes. The first concerns the detection of the ν most prominent key-points (ν = 300 [41]) and the formulation of their respective description vectors. The above are retained in two different list, viz., $\mathrm{P}_{I(t)} = \{\mathrm{p}^1_{I(t)}, \mathrm{p}^2_{I(t)}, ..., \mathrm{p}^\nu_{I(t)}\}$ and $\mathrm{D}_{I(t)} = \{\mathrm{d}^1_{I(t)}, \mathrm{d}^2_{I(t)}, ..., \mathrm{d}^\nu_{I(t)}\}$, for the points-of-interest and the descriptors respectively. Within the scope of our work, speeded-up robust features (SURF) are selected since they provide an effective balance between robustness and computational complexity [22]. These local features are intended to be used for the trajectory's segmentation and the corresponding visual words' generation. The second process regards the extraction of histogram-of-oriented-gradients (HOG) for the image's global representation [14]. These are meant for achieving image-to-image associations at the last part of our pipeline.

B. Defining sub-maps

In order to generate new sub-maps, a local feature matching coherence check is performed among consecutive image frames. This procedure is coupled with a key-points restriction step that rejects matches if their points' correspondence does not satisfy a Euclidean distance threshold α [43]. In particular, some description vectors tend to be identical to each other; however, they may represent a different point in the current view $I_{(t)}$ than the one in the just preceding frame $I_{(t-1)}$ (see top right in Fig 1). Hence, a new sub-map is determined when the correlation between the last n images' descriptors appears no more. The description vectors corresponding to each new sub-map are, subsequently, feed to a growing neural gas (GNG) clustering technique [62] for the corresponding visual words creation $\mathrm{S}_{(t)}$. Our pipeline uses GNG for local descriptors quantization since the number of extracted features is not predefined, i.e., some sequences may be shorter than others yielding a smaller amount of visual words. However, the proposed mechanism permits the

dynamic assignment of visual elements due to its incremental nature. More specifically, new vocabulary entries are generated via a frequency criterion. This way, we achieve to build a system independent from the amount of extracted local features in the incoming image and simultaneously able to perform in different environments. The maximum allowed set of generated visual words is selected to be equal with the images' extracted features ν. The parameters involved in this procedure, including the clustering iterations and the visual words generation frequency, follow the implementation in [41]. Nevertheless, when clustering is finished, local descriptors are neglected, and the database of global features $\mathrm{S}^{\mathrm{HOG}}$ is utilized along the course of navigation: $\mathrm{S}^{\mathrm{HOG}} = \{\mathrm{HOG}_{(1)}, \mathrm{HOG}_{(2)}, ..., \mathrm{HOG}_{(t)}\}$.

C. Sub-map indexing

At time t, when the latest sequence S is defined, the query process is executed for the most recently generated sub-map $\mathrm{S}_{(t)}$. Since the proposed pipeline does not adopt a global description vector for representing sequences, a voting scheme is utilized to indicate possibly loop-closure events. Visual words belonging to the query sub-map seek for the most similar ones in the database through an exhaustive k- nearest neighbor search ($k = 1$). Votes are distributed into the map, while a vote counter for each sub-map increases in agreement with its visual words pooling. The vote density $x_{\mathrm{S}}(t)$ of each database sequence S plays the most critical role in the system's confidence.

The similarity between the query and each database sub-maps is evaluated through a probabilistic score obtained by the binomial density function [61]. This way, we avoid the naïve approach of applying a heuristic threshold over $x_{\mathrm{S}}(t)$ for detecting potential loop-closures. The probabilistic score assigned to each sub-map examines the rareness of an event, i.e., high vote density in a specific sequence. This is due to the realization that when a robot is visiting a new area, which has never been encountered before, votes should be distributed randomly over the total of the already generated sub-maps, meaning that their vote density should be small:

$$X_{\mathrm{S}}(t) \sim Bin(n, p), n = N(t), p = \frac{\lambda_{\mathrm{S}}}{\Lambda(t)}. \quad (1)$$

In the above, $X_{\mathrm{S}}(t)$ represents the random variable for the number of aggregated votes of the pre-visited sub-map S at time t, N denotes the multitude of query's sequence visual words, λ is the total of visual words in S, and Λ corresponds to the size of the searching area at query time t. Finally, two conditions needs to be met before a visited sub-map is accepted. The first concerns its probabilistic score, which has to satisfy a loop-closure hypothesis threshold θ:

$$Pr(X_{\mathrm{S}}(t) = x_{\mathrm{S}}(t)) < \theta < 1, \quad (2)$$

while the second concerns the number of accumulated votes for each sub-map, which needs to be greater than the distribution's expected value:

$$x_{\mathrm{S}}(t) > E[X_{\mathrm{S}}(t)], \quad (3)$$

TABLE I: Datasets' description

Label	Sensor characteristics	# Images	Distance
KITTI 00 [63]	1241 × 376, 10 Hz	4551	12.5 Km
KITTI 02 [63]	1241 × 376, 10 Hz	4661	13.0 Km
KITTI 05 [63]	1241 × 376, 10 Hz	2761	7.5 Km
Malaga 2009 6L [64]	1024 × 768, 7.5 Hz	3474	1.2 Km
New College [65]	512 × 384, 1 Hz	2624	2.2 Km

aiming to address the rare cases in which a location gathers exceptionally few votes.

As a final note, to avoid erroneous detections originated by the robot's varying velocity, e.g., the platform remains still, a temporal window of 40s rejects recently visited areas [44]. Thanks to this mechanism, the system is endowed with the certainty that $\mathrm{S}_{(t)}$ does not share common visual features with the database.

D. Images' correspondence

Up to this point, the proposed system can highlight a similar sequence in the navigated path. However, as a final step, an image-to-image association is performed between the images belonging to the query sub-map $\mathrm{S}_{(t)}$ and the ones in candidate group S_M. Aiming to find the most similar member, cosine distance is used.

III. Measuring the performance

In this section, the evaluation protocol is presented, including the datasets, the ground truth information, and the evaluation metrics used for assessing the system's performance. Most experiments are conducted on several publicly-available datasets captured under various conditions, i.e., sensor characteristics, number of images, and different traversed distance, as shown in Table I. This way, the adaptability of our pipeline is demonstrated over several operational conditions. The first evaluation scenario comes from the KITTI vision suite [63], which constitutes a widely-known outdoor environment providing a broad range of routes with substantial loop-closure examples. As the visual stream is obtained through a stereo camera rig mounted on a forward-moving car, we considered only the left sensors' input from courses 00, 02, and 05. Similarly, the other datasets, viz., Malaga 2009 Parking 6L [64] and New College [65], have been registered by the stereo vision system of an electric buggy-typed vehicle and a robotic platform, respectively; yet, only the right monocular data are utilized for our experiments. All of the above sets contain a significant amount of loop-closures, while referring to rather different operational conditions, e.g., traveled distance and sensor frequency.

Ground truth information, which is shaped in a binary matrix of equal dimensions with the dataset's total size, is employed to measure the output of the proposed pipeline. More specifically, its boolean elements are set to 1 (ground truth$_{ij}$ = 1) for indicating an actual loop-closure event and (ground truth$_{ij}$ = 0) otherwise. For the KITTI courses, the used ground truth information was manually extracted by the

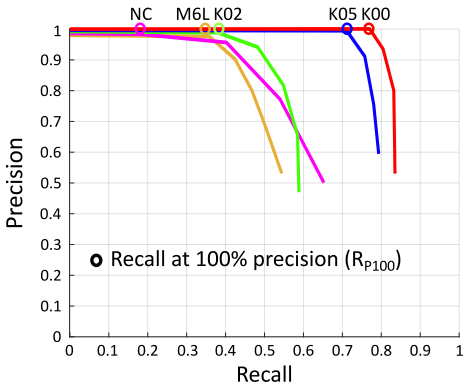

Fig. 2: Precision recall curves generated by the proposed framework. Colored cycles on the top of each graph highlight the highest recall rates for 100% precision (R_{P100}).

authors in [66]. Malaga 2009 parking 6L and New College were evaluated based on the information employed in [47].

Throughout our experimentation, the most frequently used evaluation metrics are adopted, viz., precision and recall [31]. Precision is defined as the ratio between the correct detections, i.e., true-positives, as indicated by the ground truth, over the system's total detections: true-positives / (true-positives + false positives). Recall is defined as the number of true-positives over the sum of loop-closure events in the ground truth: true-positives / (true-positives + false-negatives). Note that false-negatives are the events that ought to be detected, but the system failed to. In order to produce a single evaluation metric for the system's performance, the recall at 100% precision (R_{P100}) is utilized, which represents the highest achieved recall score without inducing any false-positive detections.

A. Performance evaluation

The system's overall performance is depicted in Fig. 2. Precision and recall curves are generated through the utilization of different loop-closure hypothesis thresholds θ. Aiming to evaluate the impact of our method, its parameters remained fixed in every tested environment. The best results at 100% of precision are indicated by the colored cycles. Our first remark is that each of the resulting curves presents high recall rate on the evaluation datasets. Evidently, the system produces a very competent performance for the KITTI 00 and 05 datasets; however, its recall drops in KITTI 02, Malaga 2009 6L, and New College, due to their intense perceptual aliasing effect.

B. System's complexity

Aiming to analyze the computational complexity, the proposed system is tested on the KITTI 00 dataset, which constitutes the most extended environment among the evaluated ones, exhibiting the highest number of loop-closures. A total of 4551 images is processed, yielding 327.6 ms per query image on average. Table II provides an extensive assessment of the system's response time. *Feature extraction* denotes the time needed for producing SURF (key-points detection and description) and HOG, while the *environment representation* process involves the timings for features' matching and visual words generation (SURF clustering) through GNG [67]. The *decision-making* step is slit into the votes distribution and the global descriptors' matching procedure. The former corresponds to the time required for the k-NN search, while the latter is the time needed for image-to-image association between the members of the query sequence and the ones belonging to the associated sub-map.

TABLE II: System's response (ms/query) for the KITTI 00 dataset [63].

		Average	Standard deviation
Feature extraction	Key-points detection	51.1	13.1
	Key-points extraction	13.6	4.6
	HOG extraction	22.6	5.1
Environment representation	SURF matching	12.1	33.7
	SURF clustering	226.0	1377.0
Decision-making	Votes distribution	45.5	30.8
	Matching	2.2	26.8
Total pipeline		327.6	1491.1

TABLE III: Comparisons against other state-of-the-art methods using the recall scores for 100% precision (R_{P100}).

Method	Dataset				
	K00	K02	K05	M6L	NC
FAB-MAP 2.0 [45]	61.2	44.3	48.5	21.8	52.6
DBoW2 [48]	72.4	68.2	51.9	**74.7**	47.5
SeqSLAM [51]	74.8	63.8	52.1	20.5	41.7
DOSeqSLAM [54]	74.8	58.9	56.7	23.3	16.8
iBoW-LCD [42]	76.5	**72.2**	53.0	57.4	**73.1**
Tracking-DOSeqSLAM [55]	77.6	61.1	38.2	42.0	40.0
Ours	**78.0**	37.3	**70.5**	35.5	17.0

As shown in Table II, loop-closures are detected efficiently, with each step achieving low computational complexity, except for the clustering process, which is the highest one. Still, this a common characteristic for every approach based on an incremental visual vocabulary. However, thanks to the small number of generated database entries describing the robot's traversed path, the time required for the votes' distribution is meager. Finally, image-to-image matching is negligibly exploiting the global descriptors' compact discrimination properties.

C. Comparative results

This section presents a comparison of the proposed pipeline against the most representative works in sequence-based mapping, namely DBoW2 [48], SeqSLAM [51], DOSeqSLAM [54], and Tracking-DOSeqSLAM [55]. Also, for the sake of completeness, comparative results are given against single-based frameworks either using a pre-trained vocabulary, such as FAB-MAP 2.0 [45], or an incremental one, such as iBoW-LCD [42]. In Table III, the recall score for flawless precision (R_{P100}) of the proposed method

TABLE IV: In depth comparison with the state-of-the-art framework in [42].

	iBoW-LCD [42]			Proposed		
	ORB(#)	S(Mb)	T(ms)	VW(#)	S(Mb)	T(ms)
K00 [63]	958K	29.2	400.2	45K	11.0	327.6
K02 [63]	950K	28.9	422.3	46K	11.2	273.3
K05 [63]	556K	16.9	366.5	25K	6.1	302.8
M6L [64]	806K	24.5	440.8	31K	7.5	253.6
NC [65]	254K	7.7	383.7	33K	8.0	102.1

and the aforementioned highly acknowledged approaches is provided. The cited methods' performance is obtained by our previous work [47], wherein each method was evaluated based on the same ground truth. Notably, the proposed system can achieve high recall rates in most environments as compared to the state-of-the-art. In KITTI courses 00 and 05, the system exhibits an improved recall score outperforming the other baseline methods. Nevertheless, in KITTI 02, Malaga 2009 6L, and New College, it performs unfavorably compared to the rest of the approaches. This is due to the absence of more sophisticated techniques, e.g., geometrical verification checks, for avoiding false-positives originating from areas with strong perceptual aliasing.

In this regard, in Table IV, we exhaustively compare the proposed pipeline with the state-of-the-art method iBoW-LCD[1] [42], which incrementally generates a visual vocabulary of binary elements, similarly to [47]. The final mapping size, i.e., visual words (VW) against ORB features [24], the storage requirements (S), and the average response time per image (T) are presented. It is worth noting that even if our method does not always imply the higher recall values, its map size (both in VW and S) and computational complexity are noticeably lower.

IV. Conclusions

In this paper, a visual sequence-based loop-closure detection pipeline is proposed. The presented method uses both local and global features for its mapping procedure, extracted from the incoming camera measurements. The former are used for the trajectory's dynamic segmentation and the corresponding visual words' generation, while the latter for image-to-image indexing. The system adopts a probabilistic scheme to find the most similar sub-map in the traversed route when querying the database. This way, an incremental visual vocabulary is constructed, offering low complexity and competitive accuracy. As evidenced by its evaluation on several publicly available datasets, our method reaches high performances while achieving a lower run-time and memory footprint as compared against a state-of-the-art method. In our future work, we intend to enhance the proposed pipeline with more sophisticated verification and indexing techniques to further increase the recall scores and reduced run-times.

[1]The iBoW-LCD [42] open-source implementation can be found at https://github.com/emiliofidalgo/ibow-lcd.

V. Acknowledgment

This work has been implemented within the project "MPU-Multirole Portable UAS" which has been financially supported by the European Regional Development Fund, Partnership Agreement for the Development Framework (2014-2020), co-funded by Greece and European Union in the framework of OPERATIONAL PROGRAMME: "Competitiveness, Entrepreneurship and Innovation 2014-2020 (EPAnEK)", Nationwide Action: "Research - Create - Innovate" (project code: T1EDK-00737).

References

[1] L. Nalpantidis, G. C. Sirakoulis, and A. Gasteratos, "Non-probabilistic cellular automata-enhanced stereo vision simultaneous localization and mapping," *Meas. Science Technology*, vol. 22, no. 11, p. 114027, 2011.

[2] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.

[3] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86–103, 2015.

[4] J. O'Keefe and D. Conway, "Hippocampal place units in the freely moving rat: why they fire where they fire," *Exp. Brain Research*, vol. 31, no. 4, pp. 573–590, 1978.

[5] E. Garcia-Fidalgo and A. Ortiz, "Vision-based topological mapping and localization methods: A survey," *Robotics and Autonomous Systems*, vol. 64, pp. 1–20, 2015.

[6] F. K. Konstantindis, A. Gasteratos, and S. G. Mouroutsos, "Vision-based product tracking method for cyber-physical production systems in industry 4.0," in *Proc. IEEE Int. Conf. Imaging Systems and Techniques*, pp. 1–6, 2018.

[7] S. Yang, S. A. Scherer, X. Yi, and A. Zell, "Multi-camera visual SLAM for autonomous navigation of micro aerial vehicles," *Robotics and Autonomous Systems*, vol. 93, pp. 116–134, 2017.

[8] F. Maffra, Z. Chen, and M. Chli, "Tolerant place Recognition combining 2D and 3D information for UAV navigation," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 2542–2549, May 2018.

[9] I. T. Papapetros, V. Balaska, and A. Gasteratos, "Multi-layer map: Augmenting semantic visual memory," in *Proc. Int. Conf. Unmanned Aircraft Systems*, pp. 1206–1212, 2020.

[10] K. Siozios, D. Diamantopoulos, I. Kostavelis, E. Boukas, L. Nalpantidis, D. Soudris, A. Gasteratos, M. Avilés, and I. Anagnostopoulos, "SPARTAN project: Efficient implementation of computer vision algorithms onto reconfigurable platform targeting to space applications," in *Proc. IEEE Int. Workshop on Reconfigurable Communication-Centric Systems-on-Chip*, pp. 1–9, 2011.

[11] E. Boukas and A. Gasteratos, "Modeling regions of interest on orbital and rover imagery for planetary exploration missions," *Cybernetics and Systems*, vol. 47, no. 3, pp. 180–205, 2016.

[12] E. Boukas, A. Gasteratos, and G. Visentin, "Introducing a globally consistent orbital-based localization system," *J. Field Robotics*, vol. 35, no. 2, pp. 275–298, 2018.

[13] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Trans. Robotics*, vol. 32, no. 1, pp. 1–19, 2015.

[14] B. Schiele and J. L. Crowley, "Object recognition using multidimensional receptive field histograms," in *Eur. Conf. Computer Vision*, pp. 610–619, 1996.

[15] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[16] A. Oliva and A. Torralba, "Building the gist of a scene: The role of global image features in recognition," *Prog. Brain Research*, vol. 155, pp. 23–36, 2006.

[17] A. Bosch, A. Zisserman, and X. Munoz, "Representing shape with a spatial pyramid kernel," in *Proc. ACM Int. Conf. Image and Video Retrieval*, pp. 401–408, July 2007.

[18] N. Sünderhauf and P. Protzel, "Brief-gist-closing the loop by simple means," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 1234–1241, 2011.

[19] H. Badino, D. Huber, and T. Kanade, "Real-time topometric localization," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1635–1642, 2012.

[20] X. Yang and K.-T. T. Cheng, "Local difference binary for ultrafast and distinctive feature description," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 188–194, 2013.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded-up robust features," in *Eur. Conf. Computer Vision*, pp. 404–417, May 2006.

[23] M. Agrawal, K. Konolige, and M. R. Blas, "CENSURE: Center surround extremas for realtime feature detection and matching," in *Proc. Eur. Conf. Computer Vision*, pp. 102–115, 2008.

[24] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Computer Vision*, pp. 2564–2571, 2011.

[25] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. Int. Conf. Computer Vision*, pp. 2548–2555, 2011.

[26] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "Kaze features," in *Proc. Eur. Conf. Computer Vision*, pp. 214–227, 2012.

[27] A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 510–517, 2012.

[28] J. P. Company-Corcoles, E. Garcia-Fidalgo, and A. Ortiz, "Towards robust loop closure detection in weakly textured environments using points and lines," in *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*, pp. 1313–1316, 2020.

[29] R. Baeza-Yates, B. Ribeiro-Neto, *et al.*, *Modern information retrieval*, vol. 463. ACM press New York, 1999.

[30] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Computer Vision*, vol. 3, pp. 1470–1470, 2003.

[31] M. Cummins and P. Newman, "Fab-map: Probabilistic localization and mapping in the space of appearance," *Int. J. Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[32] C. Mei, G. Sibley, and P. Newman, "Closing loops without places," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 3738–3744, 2010.

[33] R. Mur-Artal and J. D. Tardós, "Fast relocalisation and loop closing in keyframe-based slam," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 846–853, 2014.

[34] K. A. Tsintotas, L. Bampis, S. Rallis, and A. Gasteratos, "SeqSLAM with bag of visual words for appearance based loop closure detection," in *Proc. Int. Conf. Robotics in Alpe-Adria Danube Region*, pp. 580–587, 2018.

[35] V. Balaska, L. Bampis, M. Boudourides, and A. Gasteratos, "Unsupervised semantic clustering and localization for mobile robotics tasks," *Robotics and Autonomous Systems*, vol. 131, p. 103567, 2020.

[36] L. Bampis and A. Gasteratos, "Revisiting the bag-of-visual-words model: A hierarchical localization architecture for mobile systems," *Robotics and Autonomous Systems*, vol. 113, pp. 104–119, 2019.

[37] D. Filliat, "A visual bag of words method for interactive qualitative localization and mapping," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 3921–3926, 2007.

[38] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer, "Fast and incremental method for loop-closure detection using bags of visual words," *IEEE Trans. Robotics*, vol. 24, no. 5, pp. 1027–1037, 2008.

[39] T. Nicosevici and R. Garcia, "Automatic visual bag-of-words for online robot navigation and mapping," *IEEE Trans. Robotics*, vol. 28, no. 4, pp. 886–898, 2012.

[40] S. Khan and D. Wollherr, "IBuILD: Incremental bag of binary words for appearance based loop closure detection," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pp. 5441–5447, 2015.

[41] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "Assigning visual words to places for loop closure detection," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 5979–5985, May 2018.

[42] E. Garcia-Fidalgo and A. Ortiz, "iBOW-LCD: An appearance-based loop-closure detection approach using incremental bags of binary words," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3051–3057, 2018.

[43] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "Probabilistic appearance-based place recognition through bag of tracked words," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1737–1744, 2019.

[44] S. An, G. Che, F. Zhou, X. Liu, X. Ma, and Y. Chen, "Fast and incremental loop closure detection using proximity graphs," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 378–385, 2019.

[45] M. Cummins and P. Newman, "Appearance-only SLAM at large scale with FAB-MAP 2.0," *Int. J. Robotics Research*, vol. 30, no. 9, pp. 1100–1123, 2011.

[46] E. Garcia-Fidalgo and A. Ortiz, "Hierarchical place recognition for topological mapping," *IEEE Trans. Robotics*, vol. 33, no. 5, pp. 1061–1074, 2017.

[47] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "Modest-vocabulary loop-closure detection with incremental bag of tracked words," *Robotics and Autonomous Systems*, p. 103782, 2021.

[48] D. Gálvez-López and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.

[49] L. Bampis, A. Amanatiadis, and A. Gasteratos, "Encoding the description of image sequences: A two-layered pipeline for loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 4530–4536, 2016.

[50] L. Bampis, A. Amanatiadis, and A. Gasteratos, "Fast loop-closure detection using visual-word-vectors from image sequences," *Int. J. Robotics Research*, vol. 37, no. 1, pp. 62–82, 2018.

[51] M. J. Milford and G. F. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 1643–1649, 2012.

[52] H. Zhang, "Borf: Loop-closure detection with scale invariant visual features," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 3125–3130, 2011.

[53] K. A. Tsintotas, P. Giannis, L. Bampis, and A. Gasteratos, "Appearance-based loop closure detection with scale-restrictive visual features," in *Proc. Int. Conf. Computer Vision Systems*, pp. 75–87, 2019.

[54] K. A. Tsintotas, L. Bampis, and A. Gasteratos, "DOSeqSLAM: Dynamic on-line sequence based loop closure detection algorithm for SLAM," in *Proc. IEEE Int. Conf. Imaging Systems and Techniques*, pp. 1–6, 2018.

[55] K. A. Tsintotas, L. Bampis, A. Gasteratos, and FIET, "Tracking-DOSeqSLAM: A dynamic sequence-based visual place recognition paradigm," *IET Computer Vision*, 2021.

[56] Y. Liu and H. Zhang, "Towards improving the efficiency of sequence-based SLAM," in *Proc. IEEE Int. Conf. Mechatronics and Automation*, pp. 1261–1266, 2013.

[57] S. Garg and M. J. Milford, "SeqNet: Learning Descriptors for Sequence-based Hierarchical Place Recognition," *IEEE Robotics and Automation Letters*, 2021.

[58] S. M. Siam and H. Zhang, "Fast-seqslam: A fast appearance based place recognition algorithm," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 5702–5708, 2017.

[59] I. Kansizoglou, L. Bampis, and A. Gasteratos, "Deep feature space: A geometrical perspective," *arXiv preprint arXiv:2007.00062*, 2020.

[60] S. An, H. Zhu, D. Wei, K. A. Tsintotas, and A. Gasteratos, "Fast and incremental loop closure detection with deep features and proximity graphs," *arXiv preprint arXiv:2010.11703*, 2020.

[61] M. Gehrig, E. Stumm, T. Hinzmann, and R. Siegwart, "Visual place recognition with probabilistic voting," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 3192–3199, 2017.

[62] B. Fritzke *et al.*, "A growing neural gas network learns topologies," *Adv. Neural Information Processing Systems*, vol. 7, pp. 625–632, 1995.

[63] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012.

[64] J.-L. Blanco, F.-A. Moreno, and J. Gonzalez, "A collection of outdoor robotic datasets with centimeter-accuracy ground truth," *Autonomous Robots*, vol. 27, no. 4, pp. 327–351, 2009.

[65] M. Smith, I. Baldwin, W. Churchill, R. Paul, and P. Newman, "The New College vision and laser data set," *Int. J. Robotics Research*, vol. 28, no. 5, pp. 595–599, 2009.

[66] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, J. J. Yebes, and S. Bronte, "Fast and effective visual place recognition using binary codes and disparity information," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 3089–3094, 2014.

[67] Mostapha Kalami Heris, "Neural Gas and GNG Networks in MATLAB, Yarpiz, 2015."