Democritus University of Thrace
School of Engineering
Department of Production & Management Engineering
Laboratory of Robotics & Automation

Online Appearance-Based Place Recognition and Mapping Algorithms for Autonomous Robot Navigation

Doctoral Dissertation of:

Konstantinos A. Tsintotas

Bachelor of Science in Automation,
Technological Education Institute of Chalkida
Master of Science in Mechatronics,
Technological Education Institute of Western Macedonia

Advisory Committee:
Antonios Gasteratos, Professor (Chair)
Dimitrios Koulouriotis, Professor (Member)
Georgios Sirakoulis, Professor (Member)

Examination Committee:
Antonios Gasteratos, Professor
Dimitrios Koulouriotis, Professor
Georgios Sirakoulis, Professor
Spyridon G. Mouroutsos, Professor
Panagiotis Trahanias, Professor
Evangelos Boukas, Associate Professor
Angelos A. Amanatiadis, Assistant Professor

Xanthi, 2021

Δημοκρίτειο Πανεπιστήμιο Θράκης
Πολυτεχνική Σχολή
Τμήμα Μηχανικών Παραγωγής & Διοίκησης
Εργαστήριο Ρομποτικής & Αυτοματισμών

Αλγόριθμοι ΄Αμεσης Οπτικής Αναγνώρισης Περιοχών και Χαρτογράφησης για Αυτόνομη Πλοήγηση Ρομπότ

Διδακτορική Διατριβή:

Κωνσταντίνος Α. Τσιντώτας
Πτυχιούχος Αυτοματισμού,
Τεχνολογικό Εκπαιδευτικό ΄Ιδρυμα Χαλκίδας
Μεταπτυχιακό Δίπλωμα Ειδίκευσης Μηχανοτρονικής,
Τεχνολογικό Εκπαιδευτικό ΄Ιδρυμα Δυτικής Μακεδονίας

Τριμελής Συμβουλευτική Επιτροπή:
Αντώνιος Γαστεράτος, Καθηγητής (Επιβλέπων)
Δημήτριος Κουλουριώτης, Καθηγητής (Μέλος)
Γεώργιος Συρακούλης, Καθηγητής (Μέλος)

Εξεταστική Επιτροπή:
Αντώνιος Γαστεράτος, Καθηγητής
Δημήτριος Κουλουριώτης, Καθηγητής
Γεώργιος Συρακούλης, Καθηγητής
Σπυρίδων Γ. Μουρούτσος, Καθηγητής
Παναγιώτης Τραχανιάς, Καθηγητής
Ευάγγελος Μπούκας, Αναπληρωτής Καθηγητής
΄Αγγελος Α. Αμανατιάδης, Επίκουρος Καθηγητής

Ξάνθη, 2021

Η έγκριση διδακτορικής διατριβής από το Τμήμα Μηχανικών Παραγωγής και Διοίκησης της Πολυτεχνικής Σχολής του Δημοκρίτειου Πανεπιστημίου Θράκης, δεν υποδηλώνει την αποδοχή οποιασδήποτε γνώμης του συγγραφέα (Ν. 5343/1932, άρθρο 202, παράγραφος 2).

“True victory is victory over oneself"
-Morihei Ueshiba, founder of Aikido

Dedicated to my beloved,
Oscar
11.9.2009 - 8.4.2015

Αφιερωμένο στον αγαπημένο μου,
Όσκαρ.
11.9.2009 - 8.4.2015

Summary and Contribution

With a growing demand for autonomous robots in a wide range of applications, such as search and rescue, space, and underwater explorations, accurate navigation is more than necessary for an intelligent system to accomplish its assigned tasks. Simultaneous localization and mapping (SLAM), i.e., a robot's ability to incrementally construct a map of its working environment and subsequently estimate its position in it, has evolved over the last three decades as the core of autonomous navigation, especially when global positioning information is missing. As the importance of efficient and robust pose estimation is vital for accurate navigation, nowadays, robotics researchers have put a tremendous effort in developing methods to map the world through several exteroceptive sensors; the reason is the usefulness of an appropriate representation of the surroundings for the robot to be able to perform its tasks. However, given the sensors' noise and the absence of position measurements, even the most accurate state estimators are prone to drift inevitably accumulated over time. Hence, SLAM needs to identify when the robot revisits a previously traversed location and then to recall it. Thus, the system's drift error and uncertainty regarding the estimated position and orientation (pose) can be bounded and rectified, allowing consistent map generation. This process is widely-known as loop-closure detection and is achieved via a place recognition pipeline responsible for associating the incoming sensory data (query) with the map (database). Due to the above, the popularity of loop-closure detection in the last 30 years is not surprising if we consider the notable SLAM evolution.

Several techniques were exploited to map the robot's environment in the early years, such as range and bearing sensors, viz., lasers, radars, and sonars. However, due to the increased availability of computational power along the late years and the findings of how animals navigate using vision, mapping from optical sensors was pushed ahead. Beyond the sensor's low cost and its applicability to various mobile platforms, especially the ones with limited computational capabilities, e.g., unmanned aerial vehicles (UAVs), the main reason for cameras' utilization is related to the significant advantage of the rich textural information embedded in images, which effectively capture the environment's appearance with high distinctiveness. Not surprisingly, modern robotic navigation systems are based on appearance-based place recognition algorithms to detect loop-closures.

Nevertheless, nowadays, loop-closure detection algorithms have to provide robust navigation for an extended period. Hence, the computational efficiency and the storage requirements are vital factors for recognizing previously visited areas during long-term and large-scale SLAM operations in uncontrolled environments. The motivation behind this Ph.D. dissertation has been the prospect that in many contemporary applications, where computational resources are restricted, efficient methods are needed that can provide high performance under run-time and memory constraints. Thus, this thesis introduces several appearance-based place recognition pipelines based on different mapping techniques for addressing loop-closure detection in mobile platforms with limited computational resources. This dissertation at hand is articulated into six chapters.

Despite its unique traits, loop-closure detection is a task inextricably related to place recognition. Therefore, the dissertation would not be complete unless it briefly examines the general concept of visual place recognition in the robotics community (see Chapter 1). Similarly, a brief introduction to SLAM is provided, while the differences between the commonly used terms of localization, re-localization, and loop-closure detection are also distinguished and discussed. Finally, an overview of the currently standard problem formulation for appearance-based loop-closure detection follows, and each of its modules is reviewed in detail. This way, we provide the reader with the means to understand the contributions of our work better.

Following the previous chapter, the experimental protocol to evaluate a place recognition pipeline is presented in Chapter 2. Next, the metrics adopted for measuring the systems' performance, the collection of available datasets used throughout the experiments of this dissertation, including hand-held, car-mounted, aerial, and ground trajectories, as well as the state-of-the-art solutions taken as reference for comparisons are presented.

The third chapter introduces a novel approach for loop-closure detection based on a hierarchical decomposition of the environment. We define the generated sub-maps of the trajectory as "places" through an online and dynamic segmentation of the incoming image stream. Subsequently, when searching for candidate loops, votes are distributed to the corresponding places via an image-to-place features similarity spotting. A previously visited area is identified using a probabilistic score, while via an image-to-image pairing, the proper location match is chosen. This way, we achieve an increased system's performance, preserving at the same time the run-time low.

Following the dynamic segmentation presented in the previous chapter, in Chapter 4, we adopt a low-complexity technique for defining places in the robot's trajectory. These are generated online through a point tracking repeatability check employed on the perceived visual sensory information. When querying the database, place-to-place comparisons indicate the proper candidate sub-map and, through an image-to-image search, the appropriate location is chosen. The proposed technique reaches low timings records while keeping high performance.

Finally, the method presented in Chapter 5 exploits the advantages presented in previous methods and proposes an efficient mapping pipeline, which encodes the traversed trajectory by a low amount of unique features generated online. With this mapping technique referred to as “Bag of Tracked Words", clusters features tracked during navigation. When searching for loop-closures, probabilistic scores are assigned to every visited location, and, subsequently, the produced scores are processed through a temporal filter to estimate the belief state about the robot’s location on the map. The database’s growth rate is restricted via a management technique permitting a system with low computational complexity.

The dissertation concludes in Chapter 6. This chapter discusses and examines the current open challenges in appearance-based place recognition, e.g., map scalability for long-term operations, recognition under environmental changes, and computational complexity, which will direct our plans for extending the research described here.

Our main contributions include three different mapping techniques for addressing the task of place recognition for SLAM. The primary characteristic of each method is the absence of any previous training step preserving their online nature, i.e., throughout navigation. The first one is based on a hierarchical mapping technique with incremental clustering of the extracted visual features for achieving low-complexity with robust results. The high performance is due to a probabilistic score assigned to the database entries during query time, while the trajectory’s hierarchical representation yields to the reduced timings. The second pipeline follows the previous one regarding the dynamic image-sequence partitioning of the incoming camera stream. However, sub-maps are defined with a technique that permits the system to perform with very low timings. As a result, the proposed system can detect loop-closure events through sub-maps comparisons demonstrating sub-linear database search using almost two orders of magnitude fewer operations than other incremental-based approaches. The last pipeline exploits the positive aspects of each of the methods above to construct a robust system with high performance and low-complexity for datasets up to 13km. As the one proposed, single-based mapping techniques tend to be computationally costly, mainly due to the large number of visual elements extracted from each camera measurement. However, our framework adopts an online clustering technique for map representation, which reduces the database size intensively, producing the smallest, in terms of memory consumption and size, database up to date. Last, our method employs a Bayes filter over the robust probabilistic scores proposed in the previous pipeline, permitting higher performances under real-time constraints. As a final note, open-source implementations are made publicly available, intending to serve as a benchmark for the research community.

Περίληψη - Συμβολή στην Επιστήμη

Καθώς η ζήτηση αυτόνομων ρομποτικών συστημάτων σε ένα ευρύ φάσμα εφαρμογών αυξάνεται, όπως για παράδειγμα οι αποστολές αναζήτησης και διάσωσης, οι διαστημικές και οι υποβρύχιες εξερευνήσεις, η ακρίβεια που χρειάζεται για την πλοήγηση τους είναι απαραίτητη. Η εφαρμογή της ταυτόχρονης χαρτογράφησης και εντοπισμού της θέσης *(Simultaneous Localization and Mapping-SLAM)*, δηλαδή η ικανότητα ενός ρομπότ να δημιουργεί σταδιακά έναν χάρτη του περιβάλλοντος εργασίας του και στη συνέχεια να εκτιμά τη θέση του σε αυτόν, έχει εξελιχθεί τις τελευταίες τρεις δεκαετίες σε πυρήνα της αυτόνομης πλοήγησής του, ιδιαίτερα όταν δεν υπάρχει η πληροφορία της θέσης από το παγκόσμιο σύστημα στιγματοθέτησης. Καθώς δίνεται μεγάλη σημασία στην αποτελεσματική και εύρωστη εκτίμηση θέσης για πλοήγηση ακριβείας, οι ερευνητές στο πεδίο της ρομποτικής έχουν καταβάλει τεράστια προσπάθεια για την ανάπτυξη μεθόδων χαρτογράφησης με τη χρήση διαφορετικών εξωτερικών αισθητήρων. Ο λόγος που συμβαίνει αυτό είναι η σπουδαιότητα μιας κατάλληλης αναπαράστασης του περιβάλλοντος ώστε να μπορεί το ρομπότ να εκτελεί τις εργασίες του. Ωστόσο, λόγω του θορύβου στο σήμα των αισθητήρων και της απουσίας μετρήσεων στιγματοθέτησης, ακόμη και τα πιο ακριβή συστήματα εκτίμησης της κατάστασης του ρομπότ είναι επιρρεπή στην αναπόφευκτη συσσώρευση σφαλμάτων κατά την μετατόπιση. Ως εκ τούτου είναι σημαντικό για το σύστημα ταυτόχρονης χαρτογράφησης και εντοπισμού θέσης να βρει πότε το ρομπότ επισκέπτεται μια περιοχή που έχει προηγουμένως διασχίσει και στη συνέχεια να την ανακαλέσει. Με αυτό τον τρόπο, το σφάλμα που δημιουργείται, καθώς και η αβεβαιότητα σχετικά με την εκτιμώμενη θέση και τον προσανατολισμό του, μπορούν να περιοριστούν και να διορθωθούν, επιτρέποντας με αυτόν τον τρόπο την δημιουργία χαρτών με ακρίβεια. Αυτή η διαδικασία είναι γνωστή ως εντοπισμός κλεισίματος βρόχου *(loop-closure detection)* και επιτυγχάνεται μέσω τεχνικών αναγνώρισης περιοχών, οι οποίες είναι υπεύθυνες για τη σύνδεση των νεοεισερχόμενων δεδομένων από τα αισθητήρια (διαδικασία ερώτησης) με τον ήδη κατασκευασμένο χάρτη (βάση δεδομένων). Για τους παραπάνω λόγους, δεν αποτελεί γεγονός έκπληξης η δημοτικότητα του προβλήματος για τον εντοπισμό κλεισίματος βρόχων τα τελευταία 30 χρόνια, εάν ληφθεί υπόψη η αξιοσημείωτη εξέλιξη των συστημάτων ταυτόχρονης χαρτογράφησης και εκτίμησης

θέσης.

Τα πρώτα χρόνια, αξιοποιήθηκαν διάφορες τεχνικές για να χαρτογραφήσουν το περιβάλλον στο οποίο λειτουργεί το ρομπότ, κυρίως μέσω αισθητήρων εμβέλειας. Ωστόσο, λόγω της αύξησης της υπολογιστικής ισχύος τα τελευταία χρόνια, όπως επίσης και οι παρατηρήσεις σχετικά με το πώς πλοηγούνται τα ζώα χρησιμοποιώντας την όραση τους, ώθησαν προς τη χρήση οπτικών αισθητήρων για τη χαρτογράφηση του χώρου. Πέρα από το χαμηλό τους κόστος και την εφαρμοστικότητα τους σε διάφορες φορητές πλατφόρμες, ειδικά σε αυτές με περιορισμένες υπολογιστικές δυνατότητες, όπως τα μη επανδρωμένα εναέρια οχήματα, ο κύριος λόγος για τη χρήση των καμερών σχετίζεται με το σημαντικό πλεονέκτημα που προσφέρουν οι πλούσιες πληροφορίες που ενσωματώνονται στις εικόνες, οι οποίες αποτυπώνουν αποτελεσματικά και με υψηλή διακριτικότητα την εμφάνιση του περιβάλλοντος. Γι' αυτό τον λόγο, τα πρόσφατα συστήματα για την πλοήγηση ρομπότ βασίζονται σε αλγορίθμους αναγνώρισης περιοχών βάσει της εμφάνισης του χώρου *(appearance-based place recognition)* για τον εντοπισμό περιπτώσεων κλεισίματος βρόχου.

Ωστόσο, στις μέρες μας, οι αλγόριθμοι εντοπισμού κλεισίματος βρόχου πρέπει να παρέχουν εύρωστη πλοήγηση για μεγάλο χρονικό διάστημα. Ως εκ τούτου, η υπολογιστική αποδοτικότητα και οι απαιτήσεις χώρου για τα δεδομένα αποτελούν σημαντικούς παράγοντες κατά τη διάρκεια μακροπρόθεσμων και μεγάλης κλίμακας αποστολών σε ανεξέλεγκτα περιβάλλοντα με σκοπό την αναγνώριση επισκεπτόμενων περιοχών. Το κίνητρο για το παρόν διδακτορικό ήταν το γεγονός ότι σε πολλές σύγχρονες εφαρμογές, όπου οι υπολογιστικοί πόροι είναι περιορισμένοι, απαιτούνται αποτελεσματικές μέθοδοι που μπορούν να παρέχουν υψηλή απόδοση σε περιορισμένο χρόνο εκτέλεσης και ελάχιστη χρήση μνήμης. Έτσι, αυτή η διατριβή εισάγει μερικές τεχνικές αναγνώρισης περιοχής, οι οποίες στηρίζονται στην εμφάνιση, για διαφορετικές μεθόδους χαρτογράφησης, με σκοπό να αντιμετωπίσει το πρόβλημα του εντοπισμού κλεισίματος βρόχου σε κινητές πλατφόρμες με περιορισμένη υπολογιστική ισχύ. Έξι κεφάλαια συνθέτουν την παρούσα διατριβή.

Αρχικά, παρά τα μοναδικά χαρακτηριστικά που παρουσιάζονται στο πρόβλημα του εντοπισμού κλεισίματος βρόχου, το συγκεκριμένο αποτελεί μια διαδικασία που συνδέεται άρρηκτα με την αναγνώριση περιοχών. Επομένως, η διατριβή δεν θα ήταν πλήρης εάν δεν παρουσιάσει τη γενική έννοια της οπτικής αναγνώρισης περιοχών στην κοινότητα της ρομποτικής (δες Κεφάλαιο 1). Παρομοίως, παρέχεται μία σύντομη εισαγωγή στο σύστημα ταυτόχρονης χαρτογράφησης και εντοπισμού θέσης, ενώ διακρίνονται και συζητιούνται οι διαφορές μεταξύ των κοινώς χρησιμοποιούμενων όρων εκτίμηση θέσης, επαναπροσδιορισμός θέσης και εντοπισμός κλεισίματος βρόχου. Τέλος, ακολουθεί μια επισκόπηση σχετικά με την τυπική δομή που έχει ένα σύστημα που σχετίζεται με το πρόβλημα του εντοπισμού κλεισίματος βρόχου που βασίζεται στην εμφάνιση. Κάθε ένα από τα επιμέρους τμήματα ενός τέτοιου συστήματος εξετάζεται λεπτομερώς. Με αυτόν τον τρόπο, επιτρέπουμε στον αναγνώστη να κατανοήσει καλύτερα τις συνεισφορές της παρούσας δουλειάς.

Σε συνέχεια του προηγούμενου κεφαλαίου, στο Κεφάλαιο 2 παρουσιάζεται το πειραματικό πρωτόκολλο που απαιτείται για την αξιολόγηση μίας μεθόδου αναγνώρισης περιοχής. Έπειτα δίνονται οι μετρικές που υιοθετήθηκαν για τη μέτρηση της απόδοσης των μεθόδων, η λίστα των διαθέσιμων συνόλων δεδομένων που χρησιμοποιήθηκαν κατά τη διάρκεια των πειραμάτων αυτής της διατριβής, συμπεριλαμβανομένων διαδρομών που λήφθηκαν διά χειρός, μέσω αυτοκινήτου, καθώς και εναέριες και επίγειες διαδρομές. Επιπλέον, γίνεται εκτενή αναφορά στις μεθόδους που χρησιμοποιήθηκαν για την σύγκριση των προτεινόμενων αλγορίθμων.

Το τρίτο κεφάλαιο εισάγει μια νέα προσέγγιση για τον εντοπισμό κλεισίματος βρόχου, η οποία βασίζεται σε μια ιεραρχική αποσύνθεση του περιβάλλοντος λειτουργίας. Μέσω μίας άμεσης και δυναμικής τμηματοποίησης της εισερχόμενης ροής εικόνων, ορίζουμε τους νέους υπο-χάρτες που δημιουργούνται στην τροχιά του ρομπότ ως «μέρος» και στη συνέχεια όταν αναζητούμε υποψήφιες περιπτώσεις κλεισίματος βρόχου, κατανέμουμε ψήφους στα αντίστοιχα μέρη μέσω της ομοιότητας των χαρακτηριστικών της εικόνας-προς-το-μέρος. Μια περιοχή που το ρομπότ έχει ήδη επισκεφθεί κατά την πλοήγησή του αναγνωρίζεται μέσω ενός πιθανοκρατικού αποτελέσματος, ενώ μέσω μια εκτενούς αναζήτησης εικόνας-προς-εικόνα επιτυγχάνεται η αντιστοίχιση τοποθεσίας. Με αυτόν τον τρόπο καταφέρνουμε να έχουμε αυξημένη απόδοση διατηρώντας, ταυτόχρονα, χαμηλό χρόνο εκτέλεσης.

Ακολουθώντας τη δυναμική τμηματοποίηση που παρουσιάστηκε στο προηγούμενο κεφάλαιο, στο Κεφάλαιο 4 υιοθετούμε μία τεχνική με χαμηλή πολυπλοκότητα για τον προσδιορισμό μερών στην τροχιά που διανύει το ρομπότ. Αυτά τα μέρη δημιουργούνται άμεσα, δηλαδή κατά την διαδικασία της πλοήγησης, μέσω ενός ελέγχου της επαναληψημότητας των σημείων που εξάγονται από τα δεδομένα της κάμερας. Όταν ψάχνουμε την βάση δεδομένων για ομοιότητες, οι συγκρίσεις μεταξύ των μερών υποδεικνύουν τον κατάλληλο υποψήφιο υπο-χάρτη. Μέσω μίας τεχνικής αναζήτησης εικόνας-προς-εικόνα, τελικά επιλέγεται η κατάλληλη τοποθεσία. Η προτεινόμενη τεχνική επιδεικνύει πολύ μικρούς χρόνους εκτέλεσης ενώ διατηρεί αρκετά υψηλή απόδοση.

Τέλος, η μέθοδος που παρουσιάζεται στο Κεφάλαιο 5 εκμεταλλεύεται τα πλεονεκτήματα που παρουσιάστηκαν στα προηγούμενα κεφάλαια και προτείνει μία αποτελεσματική τεχνική χαρτογράφησης, η οποία κωδικοποιεί την πορεία του ρομπότ με μια μικρή ποσότητα μοναδικών χαρακτηριστικών που δημιουργούνται κατά την διάρκεια της πλοήγησης. Αυτή η τεχνική χαρτογράφησης, η οποία αναφέρεται ως «σάκος με ιχνηλατημένες λέξεις» *(bag of tracked words)*, ομαδοποιεί τα χαρακτηριστικά που παρακολουθούνται κατά την πλοήγηση. Κατά την αναζήτηση για ενδεχόμενο εντοπισμό κλεισίματος βρόχου, σε κάθε τοποθεσία που έχει επισκεφθεί το αυτόνομο σύστημα και στη συνέχεια αποδίδονται πιθανότητες, οι οποίες στη συνέχεια επεξεργάζονται μέσω ενός χρονικού φίλτρου για την τελική εκτίμηση της πεποίθησης που σχετίζεται με την τοποθεσία του ρομπότ στον χάρτη. Για τον περιορισμό του ρυθμού ανάπτυξης της βάσης δεδομένων εφαρμόζεται μια τεχνική διαχείρισης του χάρτη, επιτρέποντας χαμηλή υπολογιστική πολυπλοκότητα.

Τα τελικά συμπεράσματα της διατριβής παρουσιάζονται στο Κεφάλαιο 6. Σε αυτό το κεφάλαιο γίνεται συζήτηση για τις τρέχουσες προκλήσεις στο πρόβλημα της αναγνώριση περιοχών με βάση την εμφάνιση, όπως για παράδειγμα η επέκταση του χάρτη στις πολύ μακροπρόθεσμες αποστολές, η αναγνώριση περιοχών κάτω από περιβαλλοντικές αλλαγές και η υπολογιστική πολυπλοκότητα. Η εκτενής εξέταση των παραπάνω είναι αυτή που θα κατευθύνει τα σχέδιά μας για την επέκταση της έρευνας που περιγράφεται εδώ.

Οι κύριες συνεισφορές της παρούσας διατριβής περιλαμβάνουν τρεις διαφορετικές τεχνικές χαρτογράφησης για την αντιμετώπιση του προβλήματος εντοπισμού κλεισίματος βρόχου για εφαρμογές ταυτόχρονης χαρτογράφησης και εντοπισμού θέσης. Το κύριο χαρακτηριστικό που έχει κάθε μέθοδος είναι η απουσία ενός προηγούμενου βήματος εκπαίδευσης, επιτρέποντας τα να διατηρούν την άμεση φύση τους, καθιστώντας τα ανεξάρτητα καθ' όλη τη διάρκεια της πλοήγησης. Η πρώτη τεχνική βασίζεται σε μια ιεραρχική δομή χαρτογράφησης του περιβάλλοντος μαζί με σταδιακή ομαδοποίηση των εξαγόμενων οπτικών χαρακτηριστικών. Με αυτό τον τρόπο επιτυγχάνεται χαμηλή πολυπλοκότητα και ταυτόχρονα υψηλή απόδοση, η οποία οφείλεται σε ένα πιθανοκρατικό αποτέλεσμα που αποδίδεται στις προηγούμενες τοποθεσίες στην βάση δεδομένων κατά τη διάρκεια του ερωτήματος. Η δεύτερη τεχνική μας ακολουθεί την προηγούμενη σχετικά με τον δυναμικό διαχωρισμό των μερών του χάρτη. Ωστόσο, τα μέρη πλέον ορίζονται με μια μέθοδο που επιτρέπει στο σύστημα να λειτουργεί με πολύ χαμηλούς χρόνους. Αυτό έχει ως αποτέλεσμα το προτεινόμενο σύστημα να μπορεί να εντοπίσει περιπτώσεις κλεισίματος βρόχου μέσω συγκρίσεων υπο-χαρτών αποφεύγοντας την γραμμική αναζήτηση της βάσης δεδομένων και χρησιμοποιώντας σχεδόν δύο τάξεις μεγέθους λιγότερες πράξεις από άλλες προσεγγίσεις. Η τελευταία τεχνική μας εκμεταλλεύεται τις θετικές πτυχές καθεμιάς από τις παραπάνω μεθόδους ώστε να κατασκευάσει ένα εύρωστο σύστημα με υψηλή απόδοση και χαμηλή πολυπλοκότητα για διαδρομές που φτάνουν τα 13 χλμ. Όπως και η προτεινόμενη, οι μέθοδοι χαρτογράφησης που βασίζονται σε αναπαράσταση του χάρτη ξεχωριστά για κάθε εισερχόμενη εικόνα τείνουν να είναι υπολογιστικά δαπανηρές, κυρίως λόγω του μεγάλου όγκου οπτικών χαρακτηριστικών που εξάγονται από κάθε εικόνα. Ωστόσο, το σύστημα μας υιοθετεί μια άμεση τεχνική ομαδοποίησης των χαρακτηριστικών αυτών για την αναπαράσταση του χάρτη, η οποία μειώνει κατά πολύ το μέγεθος της βάσης δεδομένων, με αποτέλεσμα την

δημιουργία του μικρότερου χάρτη, όσον αφορά την κατανάλωση και το μέγεθος της χρησιμοποιούμενης μνήμης, με βάση τα σημερινά δεδομένα. Τελικά, η τεχνική μας χρησιμοποιεί ένα φίλτρο Bayes μετά τα πιθανοκρατικά αποτελέσματα, που προτάθηκαν και χρησιμοποιήθηκαν στην προηγούμενη μέθοδο μας, επιτρέποντας ακόμα πιο υψηλές επιδόσεις πάντα με γνώμονα να ικανοποιούν τους περιορισμούς πολυπλοκότητας για εφαρμογές πραγματικού χρόνου. Ως τελευταία συνεισφορά, οι παραπάνω εφαρμογές είναι ανοιχτού κώδικα και διατίθενται στο κοινό με σκοπό να χρησιμεύσουν ως σημείο αναφοράς για την ερευνητική κοινότητα.

Preface

This doctoral dissertation concentrates on the research I composed throughout the past five years as a Ph.D. student at the Democritus University of Thrace (DUTH). My research was conducted at the *laboratory of robotics and automation* (LRA), wherein I became a proud team member. During this “journey," there have been so many people who helped me, and I need to thank them. I have this feeling not because I have to but because their contribution was undeniably crucial to the overall success –if any– of this work. However, as I know that it is not possible to name each one of them, I will mention everyone who played a vital role during this journey.

This adventure began on September 16, 2015. *“Hello Professor Antonios Gasteratos, my name is Konstantinos Tsintotas. The reason behind this message is because I want to be part of your laboratory..."*. I still remember the day when I wrote my first e-mail to him. The rest is history. Five years later, I am honored to be next to him, following his guidance and his advice in every problem I confront. I thank him for being so open-hearted providing me the change to follow the difficult path of a Ph.D. student. I thank him for being there in every step of the way and making it all easy with his easygoing style. I have to admit that the past years have been the most challenging and fascinating of my life so far. He is my advisor. He will always be my dearest friend.

However, for the beginning of this journey, I owe a big "thank you" to my Master’s supervisor, prof. George F. Fragulis, who was the one that prompted me to continue my studies following the course of a doctorate. Not only he introduced me to the field of robotics through my Master’s thesis, but also he was the one who believed in me from the beginning, giving me the courage I needed for my new adventure.

During my first months as a member of the LRA, I was lucky enough to meet and learn from one senior Ph.D. candidate of our group, Dr. Loukas Bampis; now an assistant professor. He grabbed me by my newbie’s hand and passed me through all those tiring, but most important, first steps for learning the basics of our science. Moreover, we found something in common to discuss that made us to come close to each other; that was place recognition. Via our conversations, he offered me his advice and insight regarding the whole process which was lying ahead of me. He welcomed me the first time, and he is the one who stood by me as a senior researcher, and as a friend, for all this period. Without his contribution, I wouldn’t be the researcher I am today.

One year later, I had the opportunity to welcome the first woman of the group, Dr. Vasiliki Balaska. She is an outstanding researcher with whom I had the chance to work side by side for about 2 years. Throughout this period, I was impressed by her determination about her work. I cannot even begin to describe the dedication of this woman. I thank her for the beautiful moments we had together, both scientifically and personally.

Hats off to Dr. Ioannis Kansizoglou. He is the special one. In 2018, a bright youth became part of the LRA group. Since then, the man with the best mathematical mind I have ever met and the most charismatic colleague I ever had has become my best friend. Maybe it was because we were newcomers to the city of Xanthi that we found something in common to relate, making the bond of our relationship very strong. I am glad that I shared the most critical period of my studies with him, and I sincerely thank him for our long conversations that I will miss someday. This journey would not be as incredible as it was without Giannis and I am thrilled that we are now finishing this adventure together.

Furthermore, during the last two years, I had the opportunity to meet four new Ph.D. students in the LRA. The first one was Santavas Nickolaos, a hard-working man. I want to thank him for the daily long talks and plans we done together. The next one, Anastasios Taitzoglou, is the man of deeds and not of words. Our collaboration was more than professional through my difficult times in the MPU-RX4 project, and I thank him for his contribution. Looking at the rest of our new members, Tsampikos Papapetros and Fotios Konstantinidis, I can only say that our lab could not be more fortunate for engrossing such great minds. Something tells me that the three of us will cooperate more and more shortly. Moreover, I also had quite some fun supervising undergrad students. Among them, I would like to thank Panagiotis Giannis for putting up and implementing my ideas.

Those last words I owe to the people who were not able to advise on my research (even though they would love to if they could) but were always there for me to generously offer their emotional support. Within this part of beautiful people that I met at the LRA is Eleftherios Lygouras. The man with whom I shared many hours smoking and thinking about the future. He remains close to me, and still, we plan our next move. However, a special *thank you* deserves my favorite roommate, George Sapidis, who was there for me, providing day-to-day support since he moved to Xanthi. *My roommate is better than yours* is the motto that will always accompany us. Evangelos Misirlis, a junior member of the lab back in 2016, an evolving manager nowadays, was the man who help me adapt myself to the city that would host me for the next five years. I thank him from the bottom of my heart for his effort. Finally, of course, I could not exclude from this equation the wonderful colleagues Anastasia Saridou and Despoina Ioakeimidou, who were the ones I shared the two periods of COVIV-19 quarantine. I was fortunate to find those beautiful, sweet, and kind people who gave sense to my life within the pandemic. I also thank them for standing by me through the last days of my endeavor.

Needless to say, much of what I have achieved is due to my family. My father, Asterios, was the one who taught me to be patient and well-adjusted to achieve my goals, while my mother, Eleni, stood up to me during difficult periods. I will never forget my first ICRA in Brisbane, Australia in 2018. Without their support, I couldn't attend such a conference. For my little sister, Efthalia, I want to say that she is –and always will be– next to my side, and I am very proud of the kind of woman she is. To this end, for my wonderful big "sister", Ioulia Tagouti, who I love infinitely, I owe a huge *thank you* since she is my dearest friend providing me with courage during my first steps in this journey. Finally, I was lucky enough to have my beloved grandmother, Efthalia, throughout my studies. She was always there with a comforting word, brightening my life and encouraging me from the first day I began. Without her, I wouldn't be the person I am today.

Xanthi,
July 2021 *Konstantinos A. Tsintotas*

Contents

List of Figures

List of Tables

1 The revisiting problem in simultaneous localization and mapping

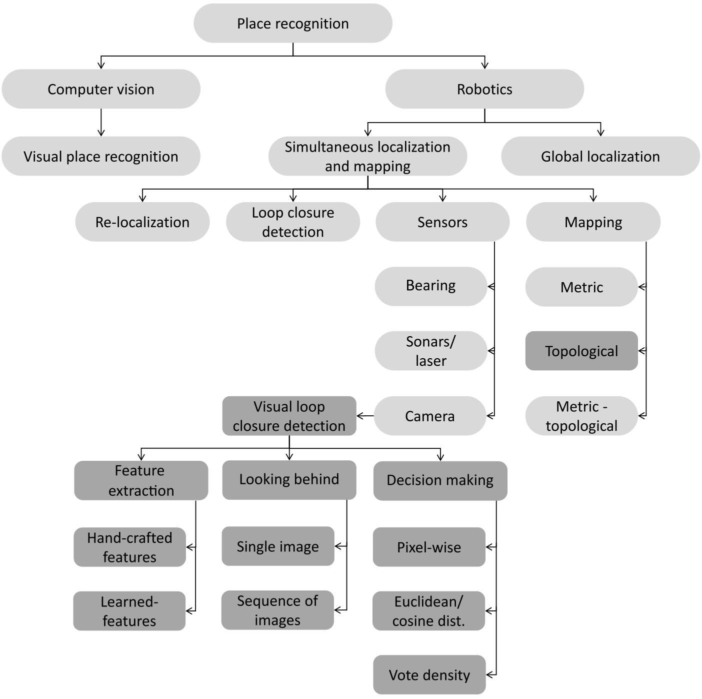


Figure 1.1. *Diagram depicting a taxonomy of place recognition in different fields. The darker colored topics are the ones described in detail within this chapter.*

Should we have been there before, we realize that viewing a single photograph is sufficient to understand where the picture was captured. This fact highlights the impact of appearance cues in localization tasks [1–3]. Historically, place recognition depicts a related task, studied intensively from the researchers in computer vision society within a broad spectrum of applications [4], including 3D reconstruction, map fusion, semantic recognition, augmented reality, and structure-from-motion. However, visual place recognition in robotics is somehow different; images under the same scene category might derive from different places [5]. Since the knowledge of an environment is a prerequisite for complex robotics tasks, place recognition is essential for the vast majority of localization implementations or re-localization and loop-closure detection pipelines within simultaneous localization and mapping (SLAM). One primary goal is that the recognizer has to generalize as much as possible, in the sense that it should support the robust association of the same place against conditional and viewpoint variations, under run-time restrictions, storage consumption, pre-training requirements, and processing power. A map diagram of the topics discussed in this chapter is depicted in Fig. 1.1. Light grey boxes indicate the topics described in brief, while darker boxes are the ones presented in detail.

1.1 Foundation of loop-closure detection

Loop-closure detection, which has long been acknowledged as the primary rectification tool in any SLAM system, historically represents a relevant and challenging task for the robotic community. Originally being introduced as “the revisiting problem," it concerns the robot’s ability to recognize whether the sensory data just captured matches with any already collected, that is, a previously visited area, aiming for SLAM to revise its position [6]. As the accumulated dead-reckoning errors in the map may persistently grow when global positioning information is not available, loop-closure detection is essential for autonomous navigation, mainly when operating in largely closed route scenarios. An important aspect is that loops inherently occur sparsely; therefore, a new observation must be added to the map if no match occurs. Since an erroneous loop-closure detection might turn out to be fatal for any SLAM framework, a reliable pipeline should detect a small number or preferably zero false-positives while still avoiding false-negatives. The former refers to situations where the robot erroneously asserts that a loop has been closed. The latter occurs when an event has been missed due to the system’s misinterpretation. Hence, “closing the loop" is a decision making problem of paramount importance for consistent map generation of unknown environments.

In the early years, several kinds of methods were exploited to map a robot’s environment, such as measuring bearings’ revolutions and range finders; however, advances were limited by the computational resources and sensor capabilities available at the time. During the last two decades, researchers can access an enviable array of sensing devices, including massively produced multi-megapixel digital cameras and computers that are more potent in processing power and storage [7]. Images, which effectively capture the environment’s appearance with high distinctiveness, are obtained through devices ranging from low-cost web cameras to high-end industrial ones. Not surprisingly, since modern robot navigation systems push towards effectiveness and efficiency, SLAM frameworks adopted such sensors and computational advances. Due to their reduced size and handiness, they can be easily attached to mobile platforms and allow the development of numerous localization and mapping pipelines with applications in different fields, such as autonomous cars [8–10], small aircrafts [11–13], and commercial devices [14].

Like any other computer vision task, visual loop-closure detection firstly extracts distinct features from images; the similarities are then calculated, and finally, confidence metrics are determined. However, vital differences exist among image classification, image retrieval, and visual loop-closure detection. More specifically, the first deals with categorizing a query image into a class from a finite number of available ones; image retrieval is contiguous to image classification and attempts to find the most relevant images in the database. On the contrary, visual loop-closure detection searches for images depicting the current robot's view while experiencing dynamic environmental and lighting conditions. Hence, a considerable interest in the community's effort has been directed towards robust image processing techniques since sensory data representations, though appropriate for classification tasks, may not perform effectively in visual loop-closure detection and vice versa.

Rather than working directly with image pixels, feature extraction techniques derive discriminative information from the recorded camera frames [15]. Hand-crafted descriptors, both global (based on the entire image) and local (based on a region-of-interest), were widely used as feature extractors. However, due to their invariant properties over viewpoint changes, local features were often selected for loop-closure detection pipelines [16]. Deep learning has revolutionized many research areas [17, 18], with convolutional neural networks (CNNs) being used for various classification tasks since they can inherently learn high-level visual features. As expected, the robotics community exploited their capabilities in loop-closure detection, especially in situations of extreme environmental changes [19]. Nevertheless, their extensive computational requirements limit their applicability in real-time applications and often induce the utilization of power-demanding general-purpose graphical processing units (GPGPUs) [20].

Through the extracted features, the system's traversed path is described by a database of visual representations. To gain confidence about its position in the map and decide whether a loop occurs, the robot needs to compute a similarity score between the query and any previously seen observation. Several techniques exist for comparing images, ranging from pixel-wise comparisons to more complex ones based on feature correspondences. Then, a similarity threshold determines if a location can be considered as loop-closure or should be declined, while additional steps, such as consistency checks based on multi-view geometry [22], can verify the matching pair. With an increasing demand for autonomous systems in a broad spectrum of applications, e.g., search and rescue [23–25], space [26, 27] and underwater explorations [28–31], the robots need to operate precisely for an extended period. As their complexity is at least linear to the traversed path, this limitation constitutes a crucial factor, severely affecting their

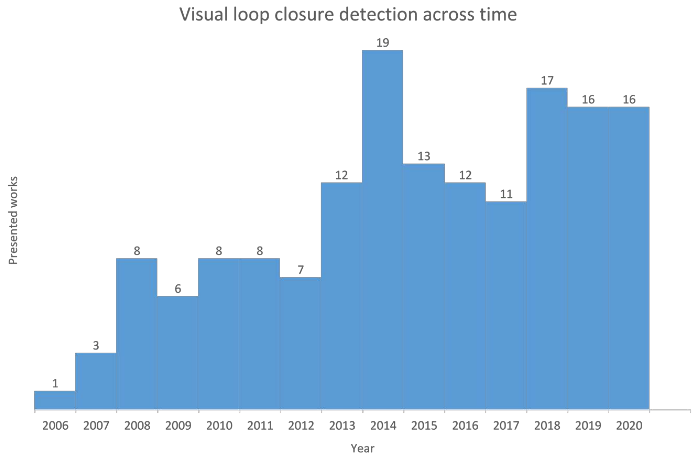


Figure 1.2. *A visual loop-closure detection evolution histogram. Starting from the first approach [21] in 2006, the growth of appearance-based place recognition systems for indicating previously visited areas in simultaneous localization and mapping applications shows that they remain a growing research field. The illustrated histogram is based on methods cited in the dissertation at hand.*

capability to perform life-long missions.

In recent years, loop-closure detection algorithms have matured enough to support continually enlarging operational environments. Thus, the research focus has shifted from recognizing scenes without notable appearance changes towards more complicated and more realistic changing situations. In such cases, detections need to be successful despite the variations in the images' content, e.g., varying illumination (daytime against night) or seasonal conditions (winter against summer). Regardless of the advancements that have been achieved, the development of systems, which are condition invariant to such changes, remains an open research field. Finally, the growing interest of the robotics community is evidenced by the number of dedicated visual loop-closure detection pipelines, as depicted in Fig. 1.2.

As we enter its third era, we need to acknowledge the groundwork laid out so far and build upon the following achievements:

1. Robust performance: loop-closure detection can operate with a high recall rate in a broad set of environments, especially when a location is revisited by a vehicle in the same direction as previously.

2. High-level understanding: loop-closure detection can extend beyond basic hand-crafted methods to get a high-level understanding and semantics of the viewing scene.

3. Data management: loop-closure detection can choose useful perceptual information and filters out irrelevant sensor data to address different tasks. Moreover, it

supports the creation of adaptive environment representations, whose complexity varies due to the task at hand.

1.2 Simultaneous localization and mapping

A robot's capability to build a map (deriving the model of an unknown environment) and localizing (estimating its position) within that map is essential for intelligent autonomous operations and, during the last three decades, one of the most famous research topics [32]. This is the classic SLAM problem, which has evolved as a primary paradigm for providing a solution for autonomous systems' navigation without depending on absolute positioning measurements, such as the ones given by global navigation satellite systems (GNSS). Nevertheless, given the noise in the sensors' signal and modeling inaccuracies, drift errors are presented even if the most accurate state estimators are used. Therefore, the robot's motion estimation degenerates as the explored environment size grows, specifically with the traversed cycles' size therein [33]. A SLAM architecture commonly comprises a front-end and a back-end component. The former handles the unprocessed sensor data modeling that is amenable for estimation, and the latter performs assumptions based on the incoming sensory inputs. Place recognition belongs to the front-end, as it is required to create constraints among locations once the robot returns to an earlier visited area [34]. In what follows, the role of loop-closure detection and re-localization in the localization and mapping engines of SLAM is analyzed, and its dependencies on the utilized sensing devices are examined.

1.2.1 Localization

Localization refers to the robot's task to establish its pose concerning a known frame of reference. More specifically, global localization examines the difficulty of recovering, in an existing map, the robot's position. At the same time, re-localization, also known as the “kidnapped-robot problem," concerns the position recovery based on a beforehand constructed map following an arbitrary “blind" displacement, viz., without awareness of the displacement, happening under heave occlusions or tracking failures. During both the above tasks, a correspondence connecting the robot's observation and a stored database representation is often known a priori. Contrariwise, loop-closure detection deals with the additional challenge of determining if the current working location belongs to a pre-visited area or not, which implies that the robot may never revisit the same mapped area, making the problem considerably more complicated [35]. However, each case is addressed by similar mechanisms using the most recent observation and a place recognizer. If a match is successful, it provides correspondence and, in many cases, a transformation matrix between the current and the database poses in the map.

1.2.2 Mapping

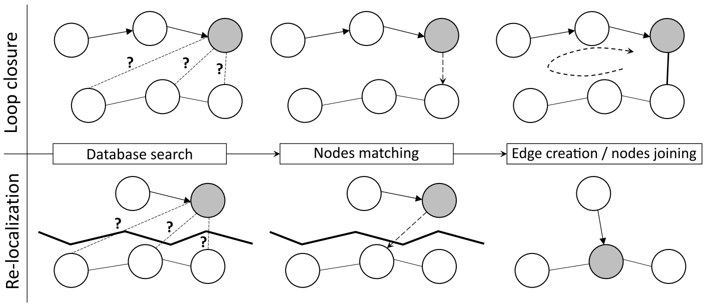


Figure 1.3. *A representative example highlighting the differences between topological loop-closure detection and re-localization. The query node (shaded observation) searches the database for candidate matches and, subsequently, the most similar is chosen. Two components are joined into one (bottom) when the system re-localizes its pose due to a tracking failure, while an edge between the two nodes is created (top) in the case of loop-closure detection.*

Trajectory mapping, which is of particular interest in autonomous vehicles, provides the robot with a modeled structure to effectively localize, navigate, and interact with its surroundings. Three major mapping models exist within SLAM, viz., metric, topological, and hybrid (metric-topological) maps. Metric maps provide geometrically accurate representations of the robot's surroundings, enabling centimeter-level accuracy for localization [36]. However, the appearance information is not considered, resulting in more frequent loop-closure detection failures in environments with repetitive geometrical structures. Moreover, this model is also computationally infeasible when large distances are dealt with [37]. Relying on a higher representation level than metric ones, topological maps mimic the humans and animals internal maps [38–40]. A coarse, graph-like description of the environment is generated, where each new observation is added as a node, corresponding to a specific location. Furthermore, edges are used to denote neighboring connections, that is, if a location is accessible from a different one. This flexible model, introduced by Kuipers and Byun [41], provides a more compact structure that scales better with the traversed route's size. Regardless of the robot's estimated metric position, which becomes progressively less accurate, these approaches attempt to detect loops only upon the similarity between sensory measurements [42–46]. In particular, two nodes become directly connected during loop-closure detection, whereas, through re-localization, two connected nodes are joined into one [47] (see Fig. 1.3). Finally, in metric-topological maps, the environment is represented via a graph-based model whose nodes are related to local metric maps, that is, a topological map is constructed, which is further split into a set of metric sub-maps [48–51].

1.2.3 Sensing

Aiming at overcoming GNSS limitations and detect loop-closures, different sensors have been used over the years, including bearing ones (e.g., wheel encoders), sonars, lasers, and cameras. Generally, range finders are chosen because of their capability to measure the distance of the robot's surroundings with high precision [52–57]. However, they are also bounded with some limitations. The sonar is fast and inexpensive but frequently very crude, whereas a laser sensor is active and accurate; however, it is slow. Within the last years, since 3D maps became more popular over traditional 2D, light detection and ranging (LiDAR) was established as the primary sensor for large-scale 3D geometric reconstructions; yet, they are unsuitable for mass installation on mobile robots due to their weight, price, and power consumption. Furthermore, its measurements, i.e., scan readings, cannot be distinguished during loop-closure detection from locations with similar shapes but different appearances, such as corridors. Although successful mapping techniques based on range-finders are implemented [58–62], these types of sensors tend to be associated with, or replaced by, single cameras [63–70] or stereo camera rigs [71–77]. This is mainly due to the rich textural information embedded in images, the cameras' low cost, and their applicability to various mobile robots with limited computational powers, such as the unmanned aerial vehicles (UAVs). Finally, even if multi-sensor frameworks [78–81] can improve performance, especially in changing environmental conditions [82], such a setup requires expensive hardware and additional calibration than camera-only ones [83]. Nowadays, autonomous robot's trajectory mapping of up to 1000 km has been successfully achieved using only cameras as the sensory modality [84].

1.3 Loop-closure detection structure

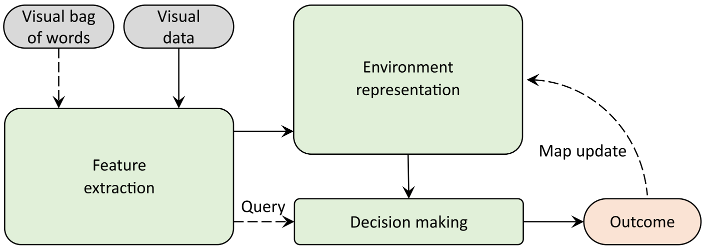


Figure 1.4. *Schematic structure depicting the essential parts of a loop-closure detection system. The image is processed to extract the corresponding visual representation, by either using trained data (visual bag of words) or not, and the robot's internal map is constructed simultaneously as the newly captured sensory measurement enters the system (visual data). When the query image arrives, its representation is compared against the database, i.e., the map, aiming to decide whether the robot navigates to an already visited area. Since loop-closures occur sparsely, the map is continually updated with new observation additions if no match occurs.*

A loop-closure detection system's generic block diagram is depicted in Fig. 1.4. Firstly, a system interpreting the environment's appearance has to detect pre-visited locations by employing only visual sensory information; thus, the perceived images have to be interpreted robustly, aiming for an informatively built map. Then, the system's internal map representation of the navigated path needs to be addressed. In many cases, such representations are driven by the robot's assigned mission. Aiming to decide whether or not the robot navigates a previously seen area, the decision extraction module performs data comparisons among the query and the database instances. Confidence is determined via their similarity scores. Lastly, as the system operates online, the map is updated accordingly throughout the autonomous mission's course. Each of the parts mentioned above is detailed in the following subsections.

1.3.1 Feature extraction

Aiming at an informative map constructed solely from visual sensing, a suitable representation of the recorded data is needed. It is not surprising that most appearance-based pipelines use feature vectors extracted from images to describe the traversed route, given their discriminative capabilities. This characteristic extends to the loop-closure detection task and renders it essential to select an effective visual feature encoder. The traditional choice for such a mechanism refers to hand-crafted features that are manually designed to extract specific image characteristics. Recently, however, the outstanding achievements in several computer vision tasks through deep learning have turned the scientific focus towards learned features extracted from CNN activations.

1.3.1.1 Hand-crafted feature-based representation

It is shown via various experimental studies that humans can rapidly categorize a scene using only the crude global information or “gist" of a scene [87,88]. Similarly, methods implemented upon global feature extractors describe an image's appearance holistically utilizing a single vector. Their main advantages are the compact representation and computational efficiency, leading to lower storage consumption and faster indexing while querying the database. However, these techniques suffer from their inability to handle occlusions, incorporate geometric information, and retain invariance over image transformations, such as those originated from the camera's motion or illumination variations. On the other hand, detecting regions-of-interest in the image and subsequently describing them has shown robustness against transformations such as rotation, scale, and some lighting variations, and in turn, allow recognition even in cases of partial occlusions. Moreover, as the local features' geometry is incorporated, they are naturally intertwined with metric pose estimation algorithms. In the last decade, most of the advances achieved in visual loop-closure detection were based on such features. An overview of both methods is illustrated in Fig. 1.5.

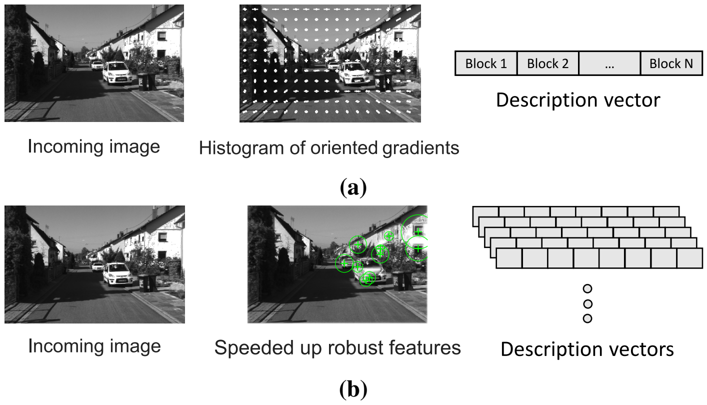


Figure 1.5. *Instances of hand-crafted feature descriptors, both (a) global (based on the entire image) and (b) local (based on regions-of-interest), extracted from the incoming image. (a) Whole image descriptors process each block in the image regardless of its context, e.g., the histogram-of-oriented-gradients [85]. (b) Local features, like the speeded-up robust features [86], are indicated in salient parts of the image and subsequently described. This way, a camera measurement is represented by the total of samples.*

Global features. Oliva and Torralba proposed the most recognized global descriptor, widely known as Gist [89–91], inspiring several loop-closure detection pipelines [92–95]. A compact feature vector was generated through image gradients extracted from Gabor filters, ranging in spatial scales and frequencies. Following the Gist's success, Sunderhauf and Protzel achieved to detect loops through BRIEF-Gist [96], a global model of BRIEF (BRIEF stands for binary robust independent elementary features [97]) local descriptor to represent the entire image. Likewise, using the speeded-up robust features (SURF) method [86], a global descriptor called WI-SURF was proposed in [83]. In [98], the authors showed that when applying disparity information on the local difference binary (LDB) descriptor [99], failures due to perceptual aliasing could be reduced.

Besides, another series of techniques for describing images globally is based on histogram statistics. Different forms, e.g., color histograms [100–102], histogram-of-oriented-gradients (HOG) [103–106], or composed receptive field histograms [107], were adopted. HOG [85], which is the most frequently used technique, calculates every pixel's gradient and creates a histogram based on the results (see Fig. 1.5a), while pyramid-of-HOG (PHOG) describes an image via its local shape and its spatial layout [108]. A differentiable version of HOG was introduced in [109]. Customized descriptors, originated from downsampled patch-based representations [110], constitute another widely utilized description method. Finally, a global descriptor derived from principal component analysis (PCA) was employed in [111].

Local features. Historically, the most acknowledged method for extracting local features is the scale-invariant feature transforms (SIFT) [112]. Based on the difference-of-gaussian (DoG) function, regions-of-interest are detected, while HOG computes their neighborhood's description. SURF (see Fig. 1.5b), inspired by SIFT, proposes a faster extraction version, while CenSurE [113], a lightweight equivalent of SURF, detects regions-of-interest using center-surrounded filters across multiple scales of each pixel's location. KAZE [114] demonstrates improved feature quality; however, it also induces higher computationally complexity. As the research community moved towards the binary description space, various feature extractors were developed offering similar SIFT and SURF performance; yet, exhibiting reduced complexity and memory requirements. Most of them extended BRIEF by incorporating descriptiveness and invariance to scale and rotation variations, such as LDB, ORB [115], BRISK [116], FREAK [117], and M-LDB [118]. Moreover, several local extractors used geometrical cues, such as line segments [119] or integrated lines and points, into a common descriptor [120], aiming to cope with region-of-interest detection in low-textured environments.

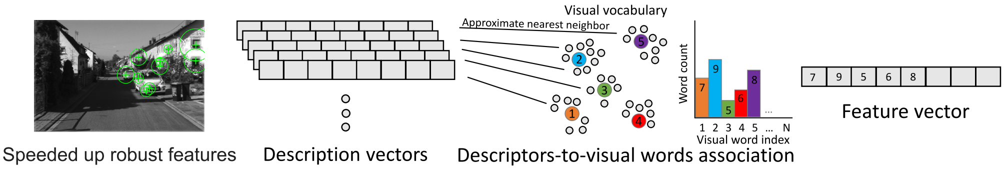


Figure 1.6. *The visual bag of words model based on a previously trained visual vocabulary. Speeded-up robust features [86] are extracted from regions-of-interest in the incoming image, and subsequently, their descriptors are connected with the most similar visual word in the vocabulary. The output vector (1 × N dimension, where N corresponds to the vocabulary's size) is a feature vector which represents the frequency of each visual word included in the camera data.*

When directly describing images using local extractors, a massive quantity of features is created [121]. This dramatically affects the system's performance, mainly when real-valued features are used [122]. Different loop-closure detection pipelines partially reduce their quantity by selecting the most informative ones [123], or utilizing binary descriptors to avoid such cases [124, 125]. Moreover, the model of the bag of words (BoW) has been employed, which was initially developed for language processing and information retrieval tasks [126], allowing the images' description as an aggregation of quantized local features, that is, "visual words" [127]. More specifically, local features are classified according to a unique database, known as "visual vocabulary," generated through unsupervised density estimation techniques [128] over a set of training descriptors (either real-valued [129, 130] or binary ones [131–133]. An overview of this process is illustrated in Fig. 1.6. However, as several visual words may occur more frequently than others, the term-frequency inverse-document-frequency (TF-IDF) scheme [134] has been adopted to weight each database element. This way, each visual word is associated with a product proportional to the number of occurrences in a given image (term frequency) and inversely proportional to its instances in the training set (inverse document frequency). Then, every image is represented via a vector of all its TF-IDF word values [135]. Fisher Kernels [136] refine the visual BoW model via fitting a Gaussian mixture over the database entries and the local features. At the same time, VLAD (VLAD stands for vector of locally aggregated descriptors [137]) concatenated the distance vectors between each local feature and its nearest visual words leading to improved performance results in the cost of increasing the memory footprint.

Although visual BoW is an efficient technique for detecting loop-closures, two weaknesses are presented. First, the visual vocabulary is typically generated a priori from training images and remains constant during navigation, which is practical; however, it does not adapt to the operational environment's attributes, limiting the overall loop-closure detection performance. Secondly, vector quantization discards the geometrical information, reducing the system's discriminative nature, primarily in perceptual aliasing cases. Consequently, several approaches address these limitations incrementally, i.e., along the navigation course, to generate the visual vocabulary [138]. This concept was introduced by Filliat [139], assuming an initial vocabulary that was gradually increased as new visual features were acquired. Similarly, Angeli *et al.* [140] merged visual words through a user-defined distance threshold. Nevertheless, most incremental visual vocabularies (either using real valued-based [141–145] or binary descriptors [146–150]) are based on the descriptors' concatenation from multiple frames to obtain a robust representation of each region-of-interest.

1.3.1.2 Learned feature-based representation

CNN is a concept introduced by LeCun *et al.* in the late '80s [151, 152]. Its deployment efficiency is directly associated with the size and quality of the training process (for place recognition, large-scale annotated datasets from a multitude of environments, such as a comprehensive set of urban areas, are needed), which generally constitute practical limitations [153]. However, its recent successes in the computer vision field are owed to a combination of advances in GPU computational capabilities and large labeled datasets [154]. The remarkable achievements in image classification [154–157] and retrieval tasks [158–160] are owed to the capability of CNNs to learn visual features with increased levels of abstraction. Hence, it was reasonable to expect that the robotics community would experiment with learned feature vectors as the loop-closure detection's backbone is oblivious to the type of descriptions used.

A fundamental question is how a trained CNN generates visual representations. To answer this, we need to consider the three following paradigms that achieve feature extraction through different processes: 1) the whole image is directly fed into a network, and the activations from one of its last hidden layers are considered as the image's descriptor [161–163]; 2) specific image regions are introduced to the trained CNN, while

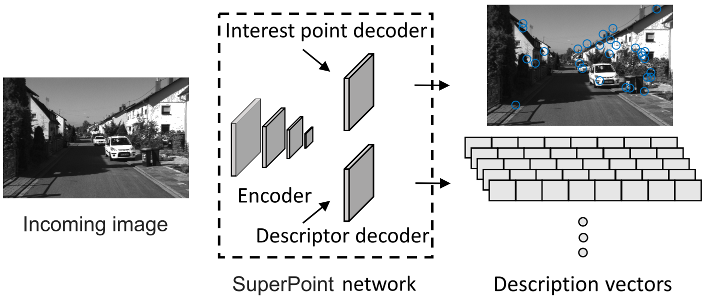


Figure 1.7. *A representative example of a fully-convolutional network that jointly extracts points of interest and their descriptors from an image.*

the respective activations are aggregated to form the final representation [164–168]; 3) the CNN receives the whole image, and via the direct extraction of distinct patterns based on the convolutional layers' responses, the most prominent regions are detected [169–176]. An illustrative paradigm is shown in Fig. 1.7. Generally, representing images globally using techniques from the first category show reduced robustness when effects, such as partial occlusion or severe viewpoint variations, are presented. Image features emerging from the second category usually cope with viewpoint changes more effectively but are computational costly since they rely on external landmark detectors. Finally, features that emerge from the third category leverage both variations, i.e., viewpoint and appearance.

Image-based features. Chen *et al.* [177] were the first to exploit learned features extracted from all layers of a trained network [161] for object recognition to detect similar locations. However, subsequent studies showed that the utilization of intermediate representations with and without the CNN's fully connected layers could offer high performances [177, 178] and rich semantic information [179–183]. Other recent contributions provided helpful insights for better understanding the complex relationship between network layers and their features visualization [184, 185]. Since then, different architectures with slight modifications have been developed and used for loop-closure detection [186–190]. Inspired by the success of VLAD, NetVLAD [162] was proposed as a trainable and generalized layer that forms an image descriptor via combining features, while the spatial pyramid-enhanced VLAD (SPE-VLAD) layer improved VLAD features by exploiting the images' spatial pyramid structure [163]. In all of the above, powerful network models were utilized as the base architecture, viz., AlexNet [154], VGG [191], ResNet [192], Inception [193], DenseNet [194], and MobileNet [195].

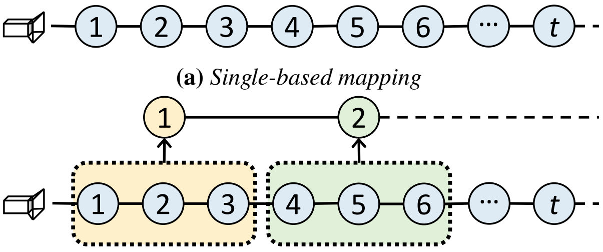
Pre-defined region-based features. Compared to the holistic approaches mentioned above, another line of works relied on detecting image landmarks, e.g., semantic segmentation and object distribution, originated from image patches to describe the visual data [196–200]. More specifically, in [164], learned local features extracted from image regions were aggregated in a VLAD fashion, while descriptors from semantic histograms and HOG were concatenated in a single vector in [197]. VLASE [198] relied on semantic edges for the image's description [201]. In particular, pixels which lay on a semantic edge were treated as entities of interest and described with a probability distribution (as given by CNN's last layer). The rest of the description pipeline was similar to VLAD. Similarly, Benbihi *et al.* presented the WASABI image descriptor for place recognition across seasons built from the image's semantic edges' wavelet transforms [200]. It represented the image content through its semantic edges' geometry, exploiting their invariance concerning illumination, weather, and seasons. Finally, a graph-based image representation was proposed in [199], which leveraged both the scene's geometry and semantics.

Extracted region-based features. The idea of detecting salient regions from late convolutional layers instead of using a fixed grid and then describing these regions directly as features have achieved impressive results [202–205]. Regions of maximum activated convolutions (R-MAC) used maxpooling on cropped areas of the convolutional layers' feature maps to detect regions-of-interest [169]. Neubert and Protzel presented a multiscale super-pixel grid (SP-Grid) for extracting features from multiscale patches [171]. Deep local features (DELF) combined traditional local feature extraction with deep learning [173]. Regions-of-interest were selected based on an attention mechanism, while dense, localized features were used for their description. SuperPoint [174] and D2-net [176] were robust across various conditional changes. By extracting unique patterns based on the strongest convolutional layers' responses, the most prominent regions were selected in [202]. Multiple learned features were then generated from the activations within each spatial region in the previous convolutional layer. This technique was additionally extended by a flexible attention-based model in [203]. Garg *et al.* built a local semantic tensor (LoST) from a dense semantic segmentation network [206], while a two-stage system based on semantic entities and their geometric relationships was shown in [204]. Region-VLAD (R-VLAD) [205] combines a low-complexity CNN-based regional detection module with VLAD. DELF was recently extended by R-VLAD via down-weighting all the regional residuals and storing a single aggregated descriptor for each entity of interest [207].

1.3.2 Looking behind

As mentioned earlier, visual localization and loop-closure detection are quite similar tasks. They share the primary goal of finding the database's most alike view, but for loop detection, all images acquired during the robot's first visit to a given area are treated as the reference set for a query view. As the system processes the sensory input data, it incrementally generates the internal map, that is, the database, which plays a vital role in the subsequent steps for location indexing and confidence estimation about its current position. Depending on how the robot maps the environment, visual loop-closure detection pipelines are distinguished into single- and sequence-based. Frameworks of the first category seek the most identical view in the robot's route, while techniques belonging in the second category look for the proper location between sub-maps, i.e., groups of individual images. This section's remainder briefly describes representative approaches by distinguishing them based on how they map the trajectory and how the system searches the database for potential matches.

1.3.2.1 Environment representation



(a) *Single-based mapping*

(b) *Sequence-based mapping*

Figure 1.8. *Depending on their trajectory mapping, appearance-based systems are divided into two main categories, namely (a) single- and (b) sequence-based. Methods of the former category represent each image in the database as a distinct location, while the latter category's schemes generate sequences, i.e., groups of individual images, along the navigation course. The observations included in each of these sequences, also referred to as sub-maps, typically consist of common visual data.*

Single-based mapping is the most common scheme for visual loop-closure detection. During navigation, the extracted visual features from each input image are associated with a specific location (see Fig. 1.8a). When the off-line visual BoW model is used, the map is formulated as a set of vectors denoting visual words at each location [129]. Otherwise, a database of descriptors indexed according to their extracted location is built [122].

In contrast to the conventional single-based methods, various frameworks use image-sequence partitioning (ISP) techniques to define group-of-images along the traversed route, which are defined as smaller sub-maps [208–211], as illustrated in Fig. 1.8b. However, many challenges emerge when splitting the map into sub-maps, such as optimal size, sub-map overlapping throughout database searching, and uniform semantic map definition [75]. SeqSLAM [110], the most acknowledged algorithm in sequence-based mapping, has inspired a wide range of authors since its first introduction [212–216]. The multitude of these pipelines, with SeqSLAM among them, uses a pre-defined quantity of images to segment the trajectory into smaller regions referred to as places.

Nevertheless, the unknown frame density, out-of-order traverses, and diverse frame separation are some of the characteristics which negatively affect the fixed-length sub-mapping methods' performance. To avoid such cases, dynamical sequence definition techniques are employed using landmarks' co-visibility properties [217–220], features' consistency among consecutive images [221], temporal models [222], or transition-based sub-mapping, e.g., through particle filtering [223].

1.3.2.2 Location indexing

A visual loop-closure detection system must search for similar views among the ones visited to decide whether a query instance corresponds to a revisited location. Firstly, database images should not share any familiar landmarks with the query. This is because images immediately preceding the query are usually similar in appearance to the recent view; however, they do not imply that the area is revisited. Aiming to prevent the system from detecting false-positives, these locations are rejected based on a sliding window defined either by a timing constant [224] or environmental semantic changes [225]. Methods based on the off-line visual BoW model employ the inverted indexing technique for searching, wherein the query's visual words indicate the locations that have to be considered as potential loop events. In contrast, methods that do not follow this model implement an exhaustive search on the database descriptors' space [226–228].

1.3.3 Decision making

The final step is the decision of whether the robot observes a previously mapped area or not. Different comparison techniques, which are broadly classified according to their map representation, have been proposed to quantify this confidence [229]; the first one is image-to-image, and the second is sequence-to-sequence. The former computes an individual similarity score for each database entry [100, 129, 226, 228], which is then compared against a pre-defined hypothesis threshold to determine whether the new image is topologically connected to the older one. Otherwise, the query cannot match any pre-visited one, resulting in a new location addition to the database. On the contrary, sequence-to-sequence is typically based on the comparison of places [67, 131, 212]. Subsequently, loop-closing image pairs are considered the groups' members with the highest similarity scores.

Moreover, to avoid erroneous detections, both temporal and geometrical constraints are employed, primarily to address perceptual aliasing conditions. Representative examples include recognizing a closed-loop only if supported by neighboring ones or if a valid geometrical transformation can be computed between the matched frames. As a final note, the resulting confidence metrics fill a square matrix whose (i, j) indexes denote the similarity between images I_i and I_j.

1.3.3.1 Matching locations

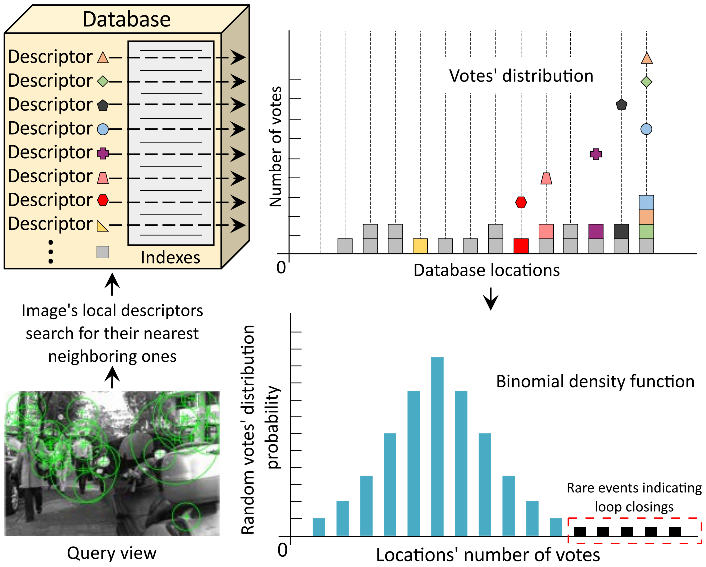

Database
Descriptor
Descriptor
Descriptor
Descriptor
Descriptor
Descriptor
Descriptor
Descriptor
Indexes
Image's local descriptors search for their nearest neighboring ones
Query view
Number of votes
Votes' distribution
0
Database locations
Random votes' distribution probability
Binomial density function
Rare events indicating loop closings
0
Locations' number of votes

Figure 1.9. *Voting procedure during query. As the most recently obtained image's local descriptors are extracted at query time, votes are distributed to database locations l from where their nearest neighbor descriptor originates. The colored and gray cubes represent the votes casted to several locations. After the locations' polling, a voting score is received that is used to evaluate the similarity. The naive approach is based on the number of votes (top-right); however, since thresholding the number of votes is not intuitive, more sophisticated methods, such as binomial density function [125], utilize the location's total amount of aggregated votes to compute a probabilistic score which highlights loop-closure detections.*

Sum of absolute differences (SAD), a location's similarity votes density, and Euclidean or cosine distance are the commonly used metrics employed to estimate the matching confidence between two instances. Directly matching the features extracted from two images represents a reasonable similarity measurement when global representations are used (either hand-crafted or learned ones). However, when local features are selected, voting schemes are selected. These techniques depend on the number of feature correspondences leading to an aggregation of votes, the density of which essentially denotes the similarity between two locations [111]. This is typically implemented by a k-nearest neighbor (k-NN) search [67, 147, 218, 226, 228, 230]. The simple approach is to count the number of votes and apply heuristic normalization [228]; however, in these cases, thresholding is not intuitive and varies depending on the environment. Rather than naively scoring the images based on their number of votes, Gehrig *et al.* [125] proposed a novel probabilistic model originated from the binomial distribution (see Fig. 1.9, bottom-right). By casting the problem into a probabilistic scheme, the heuristic parameters' effect is suppressed, providing an effective score to classify matching and non-matching locations, even under perceptual aliasing conditions. Over the years, probabilistic scores were used to enhance the system's confidence [213, 231]. Similar to the Bayes approach discussed in [7, 15], later works followed the Bayesian filtering scheme to evaluate loop-closure hypotheses [51, 84, 95]. Each of the techniques mentioned above can be efficiently adopted in sequence-based methods (e.g., SeqSLAM [110], HMM-SeqSLAM [232], ABLE-M [233], S-VWV [133], MCN [234]). By comparing route segments rather than individual camera observations, global representations are able to provide outstanding results through the utilization of relatively simple techniques. As shown in SeqSLAM, to evaluate the locations' similarity, the SAD metric is used between contrast-enhanced, low-resolution images avoiding this way the need for key-points extraction. For a given query image, comparisons between the local query sub-map and the database are performed. The likelihood score is the maximum sum of normalized similarity scores over the length of pre-defined constant velocity assumptions, i.e., alignments among the query sequence and the database sequence images. This process is inspired by speech recognition and is referred to as continuous dynamic time warping (DTW) [235]. Alignment is solved by finding the minimum cost path [236], while dynamic programming [237], graph-based optimization [215], or the incorporation of odometry information [212] strengthens its performance [232]. To improve the systems' performance, frameworks based on dynamic adjustment of the sequence length are also proposed that leverage feature matching [238], GPS priors [233], or modeling the area hypotheses over different length assumptions [239].

1.3.3.2 Exploiting the temporal consistency

In robot navigation, unlike large-scale image retrieval or classification tasks, where images are disorganized, sensory measurements are captured sequentially and without time gaps [240, 241]. Most pipelines pay a high price for indicating a loop-closure, but there is minor harm if one is missed since many chances in the following images are afforded due to the existing temporal continuity. Every sequence-based mapping technique leverages the sequential characteristic of robotic data streams aiming to disambiguate the boisterous single-based matching accuracy. The temporal consistency constraint, which is mainly adopted when single-based mapping is used, filters out inconsistent loop-closures through heuristic methods (e.g., continuous loop hypothesis before a query is accepted [147, 224]) or more sophisticated ones (e.g., the Bayesian filter [95, 129, 140, 145, 208]).

1.3.3.3 Is the current location known? The geometrical verification

After data association, a geometrical verification check is often implemented by computing a fundamental/essential matrix or other epipolar constraints. Typically, it is performed using some variation of the RANSAC (RANSAC stands for RANdom SAmple Consensus) algorithm and additionally provides the relative pose transformation if a successful correspondence is found [242]. Moreover, a minimum number of RANSAC inliers has to be satisfied for a loop to be confirmed [243, 244].

When a stereo camera rig is used [181, 245, 246], a valid spatial transformation between the two pairs of matching images is computed through the widely used iterative closest point (ICP) algorithm for matching 3D geometry [247]. Given an initial starting transformation, ICP iteratively determines the transformation among two point clouds that minimizes their points' error. Still, a high computational cost accompanies the matching process when the visual and the 3D information are combined [74, 129]. As a final note, geometrical verification is based on the spatial information of hand-crafted local features. Typically, a system that uses single vector representations (either global or visual BoW histograms) needs to further extract local features, adding more complexity to the algorithm.

1.4 Placing our contribution within the state-of-the-art

The research community has put strong foundations in the field of appearance-based place recognition to detect loops for SLAM, as the presented survey proved. To this end, we can characterize this dissertation as a missing part that fills the gap between the high performance and low-complexity needed for mobile robots within long-term operations when pre-trained techniques are avoided.

With the above notion in mind, we manage to achieve a standalone and "anytime-anywhere" ready system by adopting an online visual vocabulary formulation of the observed world using real valued-based local features upon a single-based mapping pipeline. More specifically, we exploit the positive aspects of each part in the loop-closure detection structure to construct a robust system with high performance and low-complexity for datasets up to 13km. In particular, since an efficient and robust estimation is vital for achieving accurate navigation, SURF, which provide speed and accuracy, are used. Next, an incremental visual BoW approach is utilized, as we intend to generate a visual vocabulary able to be adapted to every environment employed, avoiding this way false detections which can still arise when applied to environments with good visual characteristics which are not distinguishable due to the finite number of previously generated visual words. Through the proposed incremental-based visual BoW model, we manage to reduce the map size intensively, producing the smallest, in terms of memory consumption and size, dictionary up to date. Subsequently, the map representation constitutes a vital functionality that refers to the model followed for "remembering" the robot's knowledge about the explored world. As referred to in Section 1.3.2.1, many challenges could arise when segmenting the trajectory into places during the course navigation. To this end, the proposed system follows a singe-based mapping technique. At last, regarding the *decision making* module, following the nature of our mapping technique, i.e., online visual BoW, we use the probability density function over the accumulated votes cast by the traversed locations during query.

Nevertheless, aiming to build that system, two straightforward appearance-based place recognition methods were developed, each offering different findings. In particular, the first approach demonstrates the importance of the incremental visual BoW model in the system's performance and the probabilistic score assigned to voted locations during query. Furthermore, this method follows a hierarchical-based mapping, i.e., image-to-sequence comparisons, for achieving low-complexity. On the other hand, the second framework performs upon a sequence-based mapping technique. Aiming to take advantage of places' comparison against image-to-image, we propose a dynamic ICP method based on points' tracking. This way, our system can demonstrate sub-linear database search while at the same time preserving high accuracy.

As a final note, intending to serve as benchmarks for the research community, open-source implementations of the presented works are publicly-available. From the user's perspective, the frameworks consist of two major parts: (1) the *feature extraction* block that takes raw visual sensory data and maps the environment, and (2) the *decision making* procedure, where the system searches for loop-closure events.

2 Benchmarking

In order to benchmark a given place recognition approach, three main components are made use of: the datasets, the ground truth information, and the evaluation metrics. Accordingly to the case under study, a variety of datasets exist in the literature. The ground truth is typically formed in the shape of a boolean matrix whose columns and rows denote observations recorded at different time indices (i, j). Hence, the 1 indicates a loop-closure event between instances i and j and 0 otherwise. This matrix, together with the similarity one, is used to estimate how the system performs. Typically, the off-diagonal high-similarity elements of the generated similarity matrix indicate the locations where loops are closed. Finally, the chosen evaluation metric is the last component needed for measuring the performance.

2.1 Evaluation metrics

The relatively recent growth of the field has led to the development of a wide variety of datasets and evaluation techniques, usually focusing on precision-recall metrics [248]. These are computed from a place recognition algorithm's outcome: the correct matches are considered true-positives, whereas the wrong ones as false-positives. In particular, a correct match is regarded as any identified database entry located within a small radius from the query's location, whilst incorrect detections lie outside this range. False-negatives are the loops that had to be detected; yet, the system could not identify them. Thus precision is defined as the number of accurate matches (true-positives) overall system's detections (true-positives plus false-positives):

$$\text{Presicion} = \frac{\text{True-positives}}{\text{True-positives + False-positives}}, \tag{2.1}$$

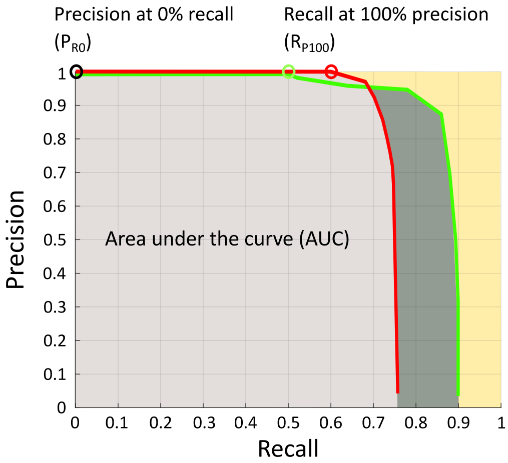


Figure 2.1. *An illustrative example of two hypothetical precision-recall curves monitoring a method's performance. A curve is extracted by altering one of the system's parameter. The highest possible recall score for a perfect precision (R_{P100}), which is the most common indicator for measuring the system's performance, is shown by the red and green cycles. The precision at minimum recall (P_{R0}) is depicted by the black cycle, while the gray color areas denote the area under the curve. At a glance, the two curves suggest that the red curve is better than the green one. Indeed, the corresponding metrics, that is, the R_{P100} at 0.6 and the expected precision at 0.8, confirm that the red curve denotes improved performance even if the area under the curve is larger in the green curve.*

whereas recall denotes the ratio between true-positives and the whole ground truth (sum of true-positives and false-negatives):

$$\text{Recall} = \frac{\text{True-positive}}{\text{True-positive + False-negatives}}. \tag{2.2}$$

A precision-recall curve shows the relationship between these metrics and can be obtained by varying a system's parameter responsible for accepting of a positive match, such as the loop-closure hypothesis threshold. The area under the precision-recall curve (AUC) is another straightforward metric for indicating the performance. Its value ranges between 0 and 1; yet, any information with respect to the curve's characteristics is not retained in AUC, including whether or not the precision reaches 100% at any recall value [249]. The average precision is also helpful when the performance needs to be described by a single value. Generally, a high precision across all recall values is the main goal for a loop-closure detection system, and average precision is capable of capturing this property. However, the most common performance indicator for evaluating a loop-closure detection pipeline is the recall at 100% precision (R_{P100}). It represents the highest possible recall score for a perfect precision (i.e., without false-positives), and it is a critical indicator since a single false-positive detection can, in many cases, cause a total failure for SLAM. However, R_{P100} cannot be determined when the generated curves are unable to reach a score for 100% precision. To overcome this problem, the extended precision (EP) metric is introduced as: EP = (P_{R0} + R_{P100}) / 2. EP summarizes a precision-recall curve through the combination of two of its most significant features, namely, precision at minimum recall (P_{R0}) and R_{P100}, into a comprehensible value. In Fig. 2.1, a representative example of two hypothetical precision recall curves is given. As shown, each depicted evaluation metric indicates that the red curve produces better performance (R_{P100} = 0.6 and EP = 0.8) than the green one; although, the area under the curve is larger in the latter hypothesis.

2.2 Datasets

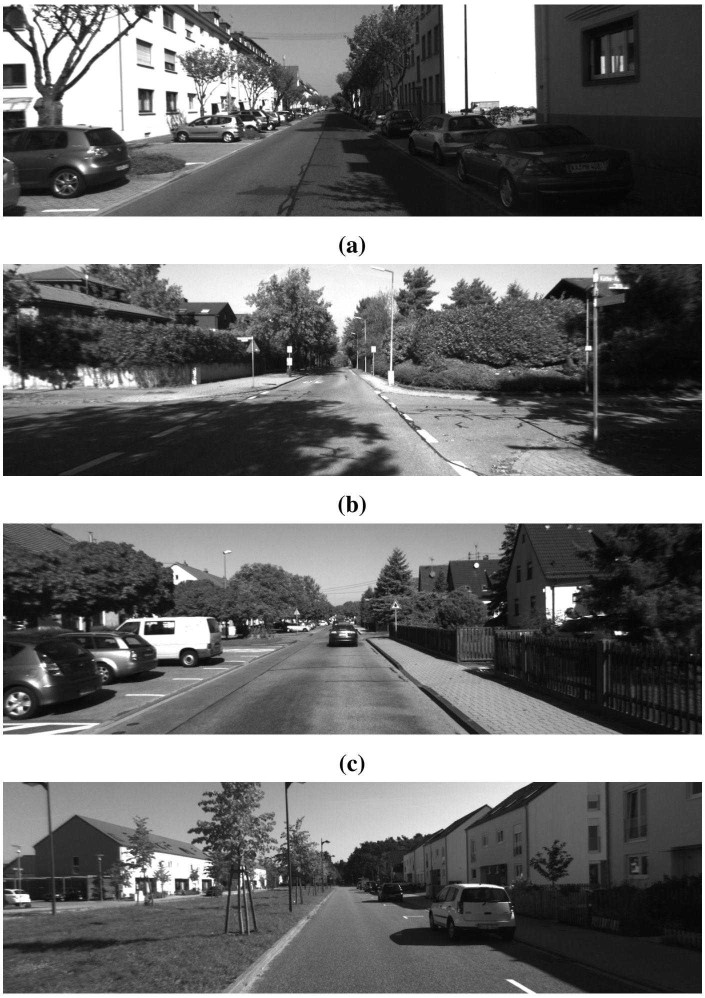
Table 2.1. *Details about the used datasets.*

Dataset	Number of frames	Traversed distance	Image size & frequency	Camera orientation
KITTI course 00	4551	≈ 12.5 km	1241 × 376, 10 Hz	Frontal
KITTI course 02	4661	≈ 13.0 km	1241 × 376, 10 Hz	Frontal
KITTI course 05	2761	≈ 7.5 km	1241 × 376, 10 Hz	Frontal
KITTI course 06	1101	≈ 3.0 km	1241 × 376, 10 Hz	Frontal
Lip 6 outdoor	1063	≈ 1.5 km	240 × 192, 1 Hz	Frontal
EuRoC MH 05	2273	≈ 0.1 km	752 × 480, 20 Hz	Frontal
Malaga parking 6L	3474	≈ 1.2 km	1024 × 768, 7 Hz	Frontal
New College	2624	≈ 2.2 km	512 × 384, 1 Hz	Frontal
Oxford City Centre	1237	≈ 1.9 km	1024 × 768, 7 Hz	Lateral

A total of nine publicly-available image-sequences are chosen for our experiments to validate the performance of the pipelines proposed in this thesis. The chosen datasets represent urban environments, indoor and outdoor areas, recorded through various platforms with a purpose to examine the systems' adaptability and capability to generalize. In Table 2.1, a summary of each image-sequence used is provided, while their detailed characteristics are discussed in the following sections.

2.2.1 Karlsruhe Institute of Technology and Toyota Technological Institute (KITTI) vision suite

A renowned benchmark environment in the robotics community is the Karlsruhe Institute of Technology and Toyota Technological Institute (KITTI) vision suite giving a wide range of trajectories (including more than 40000 images) with accurate odometry information and high-resolution image properties (for both image size and frame-rate) [250]. The suite comprises 21 image-sequences; however, the courses 00, 02, 05, and 06 are the



(a)

(b)

(c)

(d)

Figure 2.2. *Some example images that taken from the Karlsruhe Institute of Technology and Toyota Technological Institute (KITTI) vision suite [250]. From top to bottom: (a) course 00, (b) course 02, (c) course 05, and (d) course 06.*

ones used for our experiments since they provide actual loop-closure events compared to the rest ones. The incoming visual stream is captured via a camera system placed on a forward-moving car traveling through countryside roads. Nevertheless, as the proposed systems aim to monocular loop-closure detections, only one camera stream is considered. The ground truth information is not included in the KITTI suite. Thus the corresponding information is manually obtained via the dataset's odometry data by the authors in [98]. As a final note, their trajectories are plotted in the following chapters based on the pose file provided in each dataset. Example images of the used image-sequences are shown in Fig. 2.2.

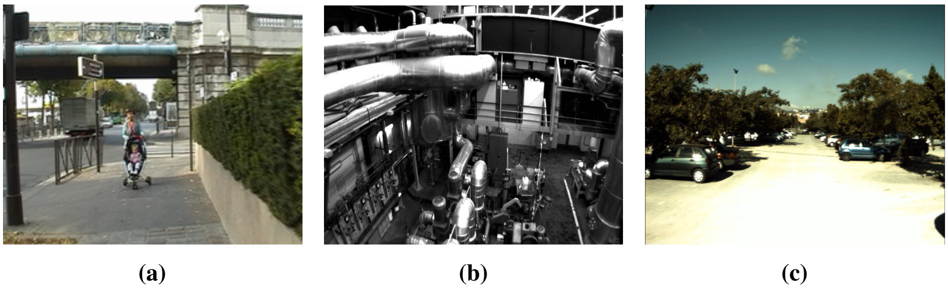


Figure 2.3. *Example images of the considered datasets. From left to right: (a) Lip6 outdoor [140], (b) EuRoC machine hall 05 [251], and (c) Malaga 2009 parking 6L [252].*

2.2.2 Lip 6 outdoor

This dataset originates from the work of Angeli *et al.* [140], providing two image-sequences. It is recorded by a handheld camera facing many loop-closures in an outdoor urban environment and a hotel corridor. Yet, for our solutions' evaluation, only the outdoor image-sequence is utilized. Both are considered challenging due to the sensor's low frame-rate and resolution, while they contain their ground truth information. What makes this dataset particularly interesting for our experiments is that areas, which are revisited along the trajectory, overlap more than two times. An illustrative example from the Lip6 outdoor image-sequence is presented in Fig 2.3a.

2.2.3 EuRoC Machine Hall 05

The EuRoC Machine Hall 05 (EuRoC MH 05), part of the EuRoC Micro Aerial Vehicle (MAV) dataset [251], is also utilized. This image-sequence presents rapid velocity variations along the trajectory and multiple examples of loop-closure events with slight fluctuations in illumination. Cameras provide visual, sensory information with a high acquisition frame rate mounted on a MAV recording an industrial environment (see Fig. 2.3b). Its ground truth information is computed manually through the corresponding highly accurate odometry data.

2.2.4 Malaga 2009 parking 6L

This environment [252] is recorded at an outdoor university campus parking lot containing mostly cars and trees, as shown in Fig. 2.3c. The camera data are provided through a stereo vision system mounted on an electric buggy-typed vehicle; however, as we aim to monocular appearance-based pipelines, only the right camera stream is used. At last, plenty of loop-closure examples are presented, while the authors in [253] manually label its ground truth.

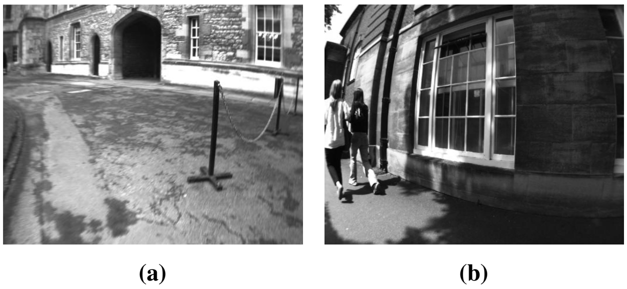


Figure 2.4. *Example images of the considered datasets. (a) New College vision suite [254] on the left side and (b) Oxford City Centre [129] on the right side.*

2.2.5 New College vision suite

This image-sequence has been registered by the vision system of a robotic platform with two cameras positioned on the left and right sides while moving through an outdoor area from the New College, University of Oxford, England [254] (see Fig. 2.4a). Yet, within the scope of this work, only the right stream is considered in our experiments. Moreover, due to the robot's low velocity and high camera frequency, its incoming visual data is resampled to 1 Hz from its initial 20 Hz rate (see Table 2.1), simulating a more representative example of modern robotic platforms. Finally, its ground truth is manually labeled through the odometry information.

2.2.6 The Oxford dataset: City Centre

City Centre belongs to the Oxford dataset, which was initially recorded for the evaluation of FAB-MAP [129]. Since then, this image-sequence is extensively utilized in visual SLAM and, in particular, to evaluate loop-closure detection pipelines. The image stream is collected by the same robotic platform used in the New College vision suite. Similarly, the right camera-input stream is selected for our experiments. City Centre is recorded to validate the ability of a system for matching images when the camera orientation is lateral (Fig. 2.4b). The authors also give ground truth information in terms of actual loop-closures.

2.3 Reference solutions

In the previous chapter, we surveyed the most important works for addressing the problem of appearance-based place recognition throughout SLAM. For this thesis, three of them are selected as reference works for validating the proposed solutions: iBoW-LCD [150], SeqSLAM [110], and the work proposed by Gehrig *et al.* [125]. More specifically, iBoW-LCD is a state-of-the-art loop-closure detection system based on an incremental BoW model for the the trajectory's mapping. Using binary features provided by ORB, the dictionary is built in a hierarchical structure for an efficient search during the query process. Aiming to reduce the uncontrolled insertion of visual elements, the authors remove visual words that are not deemed useful. Subsequently, through a temporal filter based on the concept of dynamic islands [131], the algorithm groups images close in time and prevents adjacent frames from competing among them as loop candidates. Its evaluation along the experimental protocol comes from the open-source implementation provided by the authors.

SeqSLAM constitutes one of the most recognized algorithms in sequence-based visual place recognition exhibiting the system's performance improvement by comparing group-of-images to decide about its position in the world. For this case, its open-source implementation [214] is used for our experiments. Its configuration is based on the OpenSeqSLAM2.0 MATLAB toolbox [236] except for the sequence's length, which is the most critical parameter of the algorithm. Longer sequence lengths usually perform better in terms of precision-recall, but our experiments showed the opposite behavior in some datasets. Since we want a fair comparison, this parameter is set to its default value.

The third reference work, taken into account for comparison purposes in this thesis, comes from the work of Gehrig *et al.* [125]. This approach adopts a probability score generated through the binomial density function to recognize similar previously visited areas. Based on local descriptors for image representation, the authors distribute votes to the database and then use the number of votes collected by each location. However, since a source code regarding their method is not available, we implemented a SURF-based version to offer a complete view of the impact of our solutions. For a fair comparison, its parameterization is based on the works presented in this thesis.

Nevertheless, for the sake of completeness, comparative results are also given against other representative works in visual place recognition using online BoW techniques, namely IBuILD [147], FILD [224], and Kazmi and Mertsching [95]. Moreover, for the sake of completeness, comparisons are also presented against approaches based on a previously trained vocabulary with the aim to help the reader to identify the place of the proposed pipeline within the state-of-the-art. More specifically, FAB-MAP 2.0 [84], DBoW2 [131], and PREVIeW [255] are used. The maximum recall scores achieved at 100% precision for each approach are based on the figures reported in the papers used in this thesis for image-sequences with the ground truth provided by the respective authors. The term N/A indicates that the corresponding information is not available from any cited source, while the dash (-) designates that the approach fails to reach a recall score for perfect precision. Regarding PREVIeW and FILD, evaluation occurred based on the open source implementations, with the default parameter configurations provided in the respective codes, while the authors in [95] performed the presented evaluations based on our ground truth information. In addition, for the case of FAB-MAP 2.0 and DBoW2, where no actual measurements are provided regarding the used datasets, the presented performance is obtained from the setup described by [95] and [253], respectively.

3 Probabilistic appearance-based place recognition through hierarchical mapping

As the storage requirements needed to map the whole environment in long-term applications constitute a crucial factor, in this chapter, we present an efficient appearance-based place recognition pipeline for detecting loops through the trajectory's hierarchical mapping with incremental generated visual words. The proposed pipeline, dubbed as iGNGmap-LCD, is described in detail in the following section. In Section 3.2, the experimental evaluation and the comparative results are presented.

3.1 Methodology

The proposed system operates in a pipeline fashion; the incoming data is the image stream. First, a local feature matching coherence check is performed among consecutive image frames to define the trajectory's places. Subsequently, through dynamic clustering over each sub-map's accumulated descriptors, the corresponding visual words are generated. Then, in the course of the query, local features from the most recently acquired image are extracted and seek to their most similar visual words in the database, viz., the traversed route (see Fig. 3.1). This way, a new vote is assigned to each group-of-images where the visual word corresponds, while a probabilistic score, generated through the binomial density function, determines each sub-map similarity. Next, a k-NN technique on the descriptors' space indicates the proper image in the selected place. Finally, the chosen frame is propagated to geometrical and temporal consistency checks in order to be accepted as a loop-closure match. An outline of the proposed algorithm is shown in Fig. 3.2.

3.1.1 Defining sub-maps

For each image I entering into the system, the ν most prominent key-points extracted via SURF are indicated. However, as the robot navigates through the field, some of

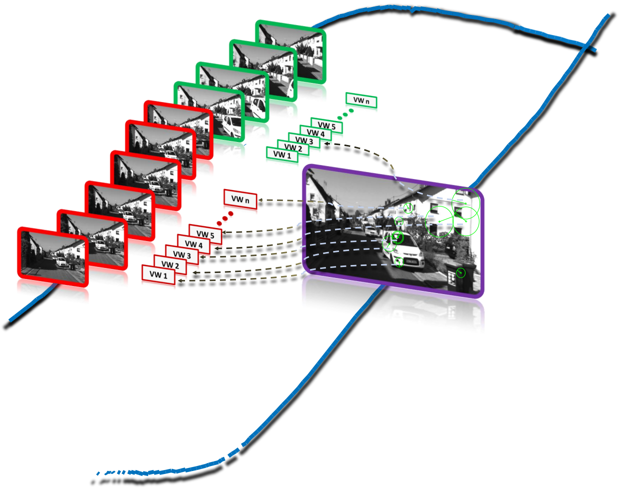


Figure 3.1. *A representation of the proposed appearance-based place recognition method while querying the database for loop-closures. Red and green outlined frames indicate different places (groups-of-images) the robot constructed during its autonomous navigation. Each of these sub-maps contains a different and unique set of incrementally generated visual words. Local feature descriptors seek their nearest neighboring visual words at query time, distributing votes to previously visited sub-maps. The proper pair (image-to-place) is indicated through the probabilistic score generated via binomial density function upon each group's vote aggregation (density of dashed arrows).*

the incoming camera measurements may not produce enough visual information, e.g., observing a black plain. Therefore, to avoid the definition of inconsistent sub-maps, images that contain less than ξ key-points are rejected. To this end, during the robot's online operation, the projected descriptor space is constantly updated by the detected feature vectors d_I. Yet, instead of reducing the descriptors' dimensionality, the proposed algorithm utilizes the whole SURF space.

More specifically, new sub-maps S are determined through a feature matching coherence check. This way, at time t, the incoming image stream $I_{(t-n)}$, ..., $I_{(t-2)}$, $I_{(t-1)}$, $I_{(t)}$ is segmented when the correlation between the last n images' descriptors cease to exist:

$$\left| \bigcap_{i=0}^{i=n} d_{I_{(t-i)}} \right| \leq 1 \tag{3.1}$$

where $|X|$ denotes the cardinality of set X. A descriptors' database D_S is also retained for each group-of-images, as shown in Fig. 3.2 (left), via:

$$D_S = \bigcup_{i=0}^{i=n} d_{I_{(t-i)}} \tag{3.2}$$

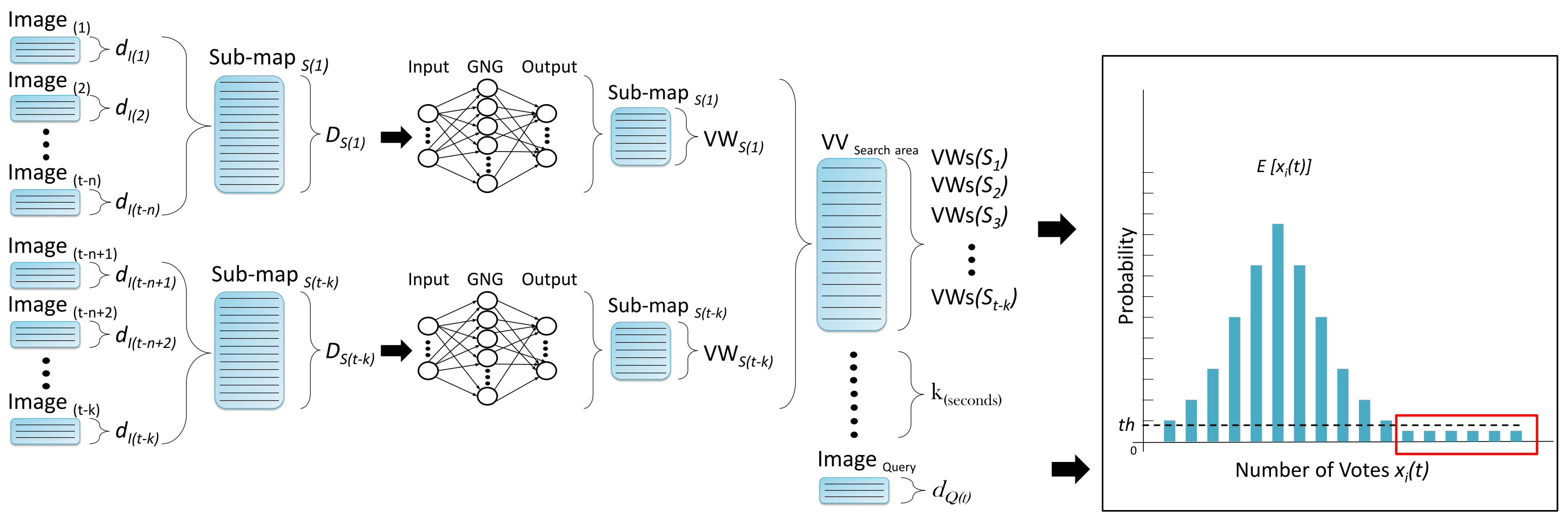


Figure 3.2. *An overview of the proposed appearance-based place recognition pipeline for loop-closure detection. As the incoming image stream enters the pipeline, dynamic places, defined as sub-maps ($S(t)$), are formulated through a feature matching technique (left). Subsequently, the accumulated local feature descriptors ($D_{S(t)}$) are fed into the growing neural gas (GNG) clustering mechanism [256], where the corresponding sub-map's visual words (VWs(S_i)) are generated (center). At query time, image descriptors ($d_{Q(t)}$) seek their nearest neighboring visual words into the incrementally generated visual vocabulary (VV). During descriptors-to-visual words matching, votes are shared among the places, where the latter belong. Finally, the candidate place is located according to its probabilistic score originated by its vote density (right). The highlighted red area indicates the density rarity a loop-closure event would produce.*

3.1.2 Assigning visual words to places

In order to assign visual words to sub-maps, local descriptors D_S are utilized as input to the growing neural gas clustering algorithm [256]. In contrast to other popular clustering methods, where the number of clusters is predefined, the proposed clustering method incrementally adds new nodes, i.e., visual words, until an error-minimization criterion is met. Since our approach uses the specific mechanism for quantizing the feature vectors, its main parameterization remains the same as the original implementation. The maximum allowed set of visual words (α), generated by the growing neural gas, is determined as equal to the images' extracted local features ν. This analogy is chosen to provide a direct correspondence between visual words and image features. Thus, a new visual word is created when a frequency criterion φ is met, defined as the ratio between the maximum number of visual words per place and the mean of places' length μ ($\varphi = \nu / mean(\mu)$). At the same time, as the system intends to be of low computational complexity, the number of iterations (ε) needed for the clustering mechanism is selected to the lowest permissible. Finally, a visual vocabulary, that is, the incremental constructed database, is retained during the procedure:

$$\mathrm{VV} = \bigcup_{i=1}^{i=t} \mathrm{VWs}(S_i), \tag{3.3}$$

where the term S_t is the latest formulated sub-map in the trajectory (see Fig. 3.2 center). An indexing list is also maintained along the visual vocabulary providing image-to-place associations during the inference procedure.

3.1.3 Sub-map indexing

Aiming to avoid false-positive loop-closure detections originating from early visited locations, a searching area ($\mathrm{VV}_{\text{Search area}}$ in Fig. 3.2) that rejects recently acquired input images is defined based on a temporal constant ψ:

$$\mathrm{VV}_{\text{Search area}} = \mathrm{VV} \cap [\mathrm{VWs}(S_1), \mathrm{VWs}(S_{t-\psi})] \tag{3.4}$$

Given a query image I_Q, a searching procedure is performed among the produced sub-maps to detect loop-closure candidates. The nearest neighbor mechanism projects the query's local features to the visual words included to the search area. Unlike most previously trained visual BoW-based systems, where histogram comparison techniques are used, the proposed method adopts a voting scheme. This way, the current image's feature descriptors seek the most similar visual words into the database, and votes are distributed to places according to the visual words' origin (see Fig. 3.3). The vote density $x_i(t)$ of each place i constitutes the factor for determining the probabilistic score. It is worth noting that even if a threshold over the accumulated votes could be

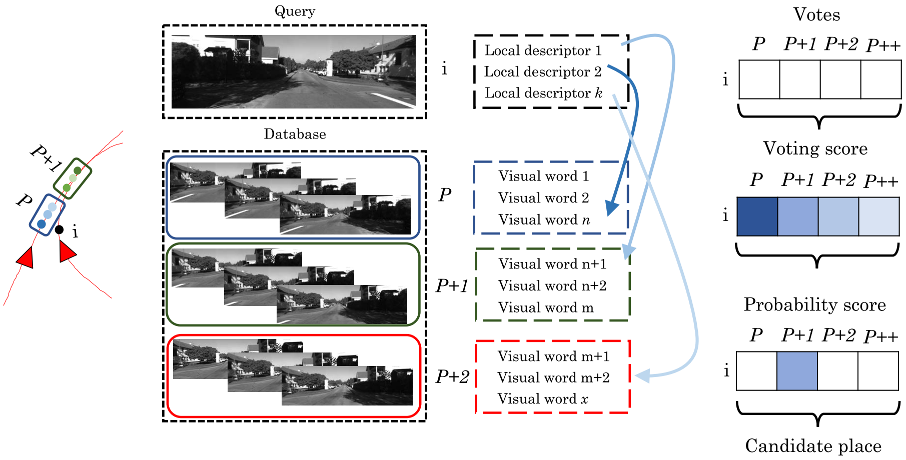


Figure 3.3. *The query process of the proposed single-based hierarchical mapping visual place recognition method. As the incoming image stream is processed, votes are distributed to sub-maps based on the local feature descriptor to visual word association. Subsequently, the candidate sub-map is indicated through the probabilistic score, given via the binomial density function over its accumulated votes.*

applicable, adopting such a naïve technique is uncertain how the system would behave when the number of votes is insufficient (e.g., due to low textured visual information).

Hence, a binomial probability function is employed to check the trajectory for potential revisited areas when the voting procedure is completed. More specifically, the nature of the binomial function is to seek for rareness events. In particular, in cases where the robot traverses a hitherto unseen location (never encountered before), votes should be randomly distributed to their nearest neighboring words in the database even if they are not accurately associated with a similar one. This fact constitutes a common event with high probability, meaning that the locations' vote density should be low. Ergo, the number of aggregated votes for each database place should obey a binomial distribution (see equation 3.5). Contrariwise, when confronting a previously visited environment, the corresponding votes cast for a specific location increase. Thus, the random vote distribution expected from the binomial function would be violated. As a consequence, the event would be considered of low probability with an increased voting score. Such instances are interpreted as loop-closure candidates by the proposed system:

$$X_i(t) \sim Bin(n,p), n = N(t), p = \frac{\lambda_i}{\Lambda(t)}, \tag{3.5}$$

where N denotes the multitude of query's local feature descriptors (d_Q), λ corresponds to the total of database sub-map's i visual words, and $\Lambda(t)$ is the total of visual words within the searching area ($\text{VV}_{\text{Search area}}$). The probability score is calculated for each place, while two conditions have to be satisfied before a candidate is recognized as known. Its score has to satisfy a threshold value th:

$$Pr(X_i(t) = x_i(t)) < th < 1, \tag{3.6}$$

while, the number of accumulated visual words for the specific sub-map needs to be greater than the distribution's expected value:

$$x_i(t) > E[X_i(t)]. \tag{3.7}$$

The second condition is responsible for discarding the cases where fewer votes are collected via random voting.

3.1.4 Images' correspondence

Up to this point, iGNGmap-LCD has a strong belief about a previously visited place in the traversed route. As a final step, an image-to-image correlation is performed between the current image and the most similar member of the chosen sub-map $S_{(m)}$ in the database. Based on a k-NN classifier ($k = 1$), the query's descriptors d_Q are matched with those ($D_{S_{(m)}}$) belonging to $S_{(m)}$. The image that gathers the most matches is considered a loop-closure candidate and proceeds for further validation.

In particular, the chosen pair is subjected to a geometrical consistency check to avoid a false-positive match. Aiming to achieve this, we try to estimate a fundamental matrix T between the selected images using a RANSAC-based scheme. If T computation fails or the number of inlier points between the two images is less than a factor τ, the candidate image is ignored. Finally, intending to accept a matching pair, the method incorporates a temporal consistency check among the last β input frames. More specifically, a loop-closure event is accepted when the aforementioned conditions are met for β consecutive images.

3.2 Experimental results

This section evaluates the proposed system through several experiments and compares the achieved performance and storage consumption needed against the reference method of iBoW-LCD reported in chapter 2. Moreover, performance comparisons are given against other place recognition pipelines, namely FAB-MAP 2.0 and DBoW2. The method is tested upon five image-sequences: KITTI courses 00, 02, 05, Malaga parking 6L, and New College. However, only KITTI course 05 is used for the parameters' evaluation, while the others for measuring the system's performance. In all cases, the algorithm is configured using the values indicated in Table 3.1. These parameters remain fixed in all tested scenarios aiming to assess the iGNGmap-LCD's impact. The average place' length μ is observed to be approximately 12 for most of the tested datasets during

Table 3.1. *Parameters utilized from the proposed hierarchical mapping pipeline.*

Parameter	Symbol	Value
Minimum detected local features per image	ξ	5
Maximum prominent local features per image	ν	300
Maximum generated visual words per place	α	300
Search area time constant	ψ	40 sec
Geometrical verification inliers	τ	12 [131]
Images' temporal consistency	β	2
Probability score threshold	σ	10^{-12}

the experiments. In addition, the chosen searching offset ψ is selected accordingly to avoid detections with strong spatiotemporal relationships.

3.2.1 Performance evaluation

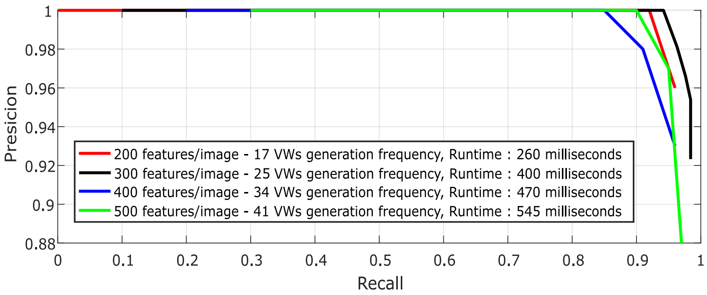


Figure 3.4. *Precision-recall curves evaluating the utilized local features ν per image. This corresponds to the number of visual words generated in each place (α). 300 features per image provide better recall rate, while the execution time remains low. Experiments are performed on KITTI course 05 [250].*

As the loop-closure threshold th is varied, we monitor the precision-recall curves obtained for different cases of local feature number preserved for every image (ν = 200, 300, 400, 500). As shown in precision-recall curves depicted in Fig. 3.4, the system's achieved performance resembles in cases where the extracted elements are less than 400. As the number of accepted features increases, it is observed that the recall rate (corresponding to 100% precision) is decreasing. This is because weak features, detected during the robot's first visit to a specific area, are mainly noisy. Thus, it is less probable for them to be matched during a loop-closure event.

In addition, we assess the effect of growing neural gas iterations in the execution time needed for generating the corresponding visual words. As illustrated by the red curve in Fig. 3.5, the system's performance improves when the number of iterations increases. However, the execution time is raised by a factor of two from the first to the second

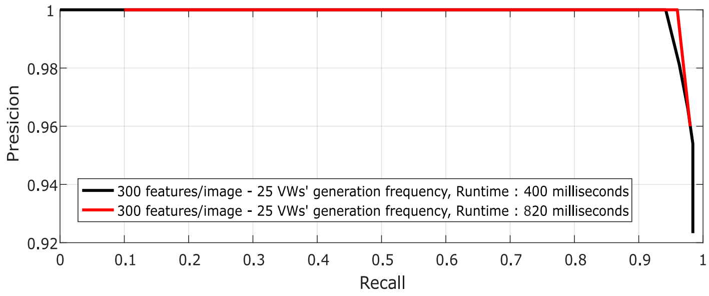


Figure 3.5. *Precision-recall curves evaluating the execution time of the proposed algorithm against the growing neural gas iterations. While the second iteration (red line) doubles the complexity, the recall rate (for 100% precision) indicates a similar performance to the first one (black line). Experiments are implemented on KITTI course 05 [250].*

iteration (from 400 ms to 820 ms). Bearing that in mind, a small percentage of recall is sacrificed for a faster implementation. The overall performance of iGNGmap-LCD is shown in Fig. 3.6.

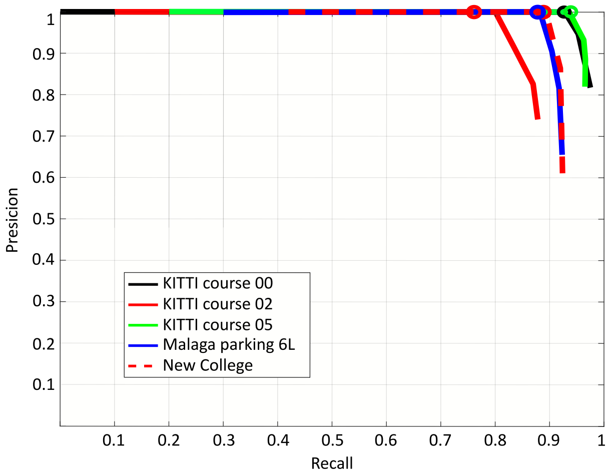


Figure 3.6. *Precision-recall curves for the proposed approach. Color markers (cycles) on the top of the graphs highlight the highest recall for* 100% *precision (*R_{P100}*).*

3.2.2 System's response

Table 3.2. *Processing time per image (ms/query) of iGNGmap-LCD.*

		Average Time (ms)
Feature extraction	Key-point detection Key-point description	51.1 13.6
Environment representation	SURF matching SURF clustering	12.1 226.0
Decision making	Votes distribution Matching	45.5 54.5
Total pipeline		402.8

Similarly, aiming to analyze the computational complexity, the proposed system is tested on KITTI course 05. In Table 3.2, an extensive assessment of the system's response time is provided. *Feature extraction* denotes the time needed for extracting SURF (key-points detection and description), while the *environment representation* process involves the timings for feature matching and visual words' generation (SURF clustering) through growing neural gas. The *decision making* step is slit into the votes' distribution and the image-to-image matching procedure. The former corresponds to the time required for the k-NN search, while the latter is the time needed for descriptors' association between the members of the query place and the ones belonging to the query image. The time for the geometrical verification is also included. As shown in Table 3.2, loop-closures are detected efficiently, with each step achieving low-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 also fast even if a geometrical check is performed.

3.2.3 Comparative results

Table 3.3 compares the recall score for flawless precision (R_{P100}) of the proposed method against two well-known approaches. The results show that iGNGmap-LCD can achieve high recall rates in every tested environment outperforming each of the systems with which it is compared. In addition, in Table 3.4, we exhaustively compare our pipeline with the reference method of iBoW-LCD. The final mapping size, i.e., the visual vocabulary of SURF and ORB, the storage requirements S(Mb), and the recall scores R(%), are presented. It is worth noting that our method implies the higher recall

Table 3.3. *Performance comparison with other well-known appearance-based place recognition methods. The results show the achieved recall rates (%) for 100% precision (R_{P100}). Entries highlighted with bold indicate the best performing approach for each dataset.*

	FAB-MAP 2.0	DBoW2	iGNGmap-LCD
Dataset	R(%)	R(%)	R(%)
KITTI course 00	61.2	72.4	**93.1**
KITTI course 02	44.3	68.2	**76.0**
KITTI course 05	48.5	51.9	**94.2**
Malaga parking 6L	21,8	74.7	**87.9**
New College	52.6	47.5	**88.0**

Table 3.4. *In depth comparison with the framework of iBoW-LCD. Number of generated visual words SURF(#), vocabulary storage consumption S(Mb), and recall score R(%) for the proposed pipeline and iBoW-LCD are given. As can be observed, our system achieves substantially lower amount of generated visual words for each evaluated dataset, while reaching high performance for every evaluated image-sequence.*

	iBoW-LCD			iGNGmap-LCD		
Dataset	ORB(#)	S(Mb)	R(%)	SURF(#)	S(Mb)	R(%)
KITTI course 00	958K	29.2	76.5	**45K**	**11.0**	**93.1**
KITTI course 02	950K	28.9	72.2	**46K**	**11.2**	**76.0**
KITTI course 05	556K	16.9	53.0	**25K**	**6.1**	**94.2**
Malaga parking 6L	806K	24.5	57.4	**31K**	**7.5**	**87.9**
New College	254K	7.7	73.1	**33K**	**8.0**	**88.0**

values, while at the same time, its map size (both in terms of generated visual words and memory consumption) is noticeably smaller.

4 Dynamic places' definition for sequence-based visual place recognition

As mentioned in Chapter 1, many challenges arise when breaking the map into sub-maps. Using a predefined number of images and a sliding window scheme improves a sequence-based visual place recognition pipeline; however, this functionality is computationally costly since the robot needs to compare its query place with every possible sub-map that not exhibits the same semantics as its neighboring ones.

Aiming for an efficient and low-complexity framework independent from any training procedure, we present Tracking-DOSeqSLAM. The proposed paradigm relies on local point tracking for dynamic and online sub-map definition. Points are extracted from the perceived camera measurement, and through the Kanade-Lucas-Tomasi (KLT) tracker [257], a new place is determined when the contained point tracking fails to advance in the next image frame. This way, we avoid the computationally costly feature matching process among consecutive images. a technique frequently adopted for ICP, while robustness is achieved regarding the generated places' size. The remainder of this chapter is organized as follows. In Section 4.1, the proposed pipeline is described in detail, while in Section 4.2, the proposed framework is evaluated in seven datasets and compared in depth with its reference approach, i.e., SeqSLAM, as well as our solution presented in Chapter 3.

4.1 Methodology

This section presents an extended description of the proposed low-complexity and sequence-based place recognition pipeline. Since the proposed algorithm aims at the sequences' dynamic definition, the transition from the fixed-size approach to the proposed version is presented. First, based on local key-points extracted from the incoming image, the system formulates each place dynamically through the KLT point tracker. Subsequently, following the SeqSLAM's feature extraction steps, the visual data are downsampled and normalized. Next, each image is compared to each previously

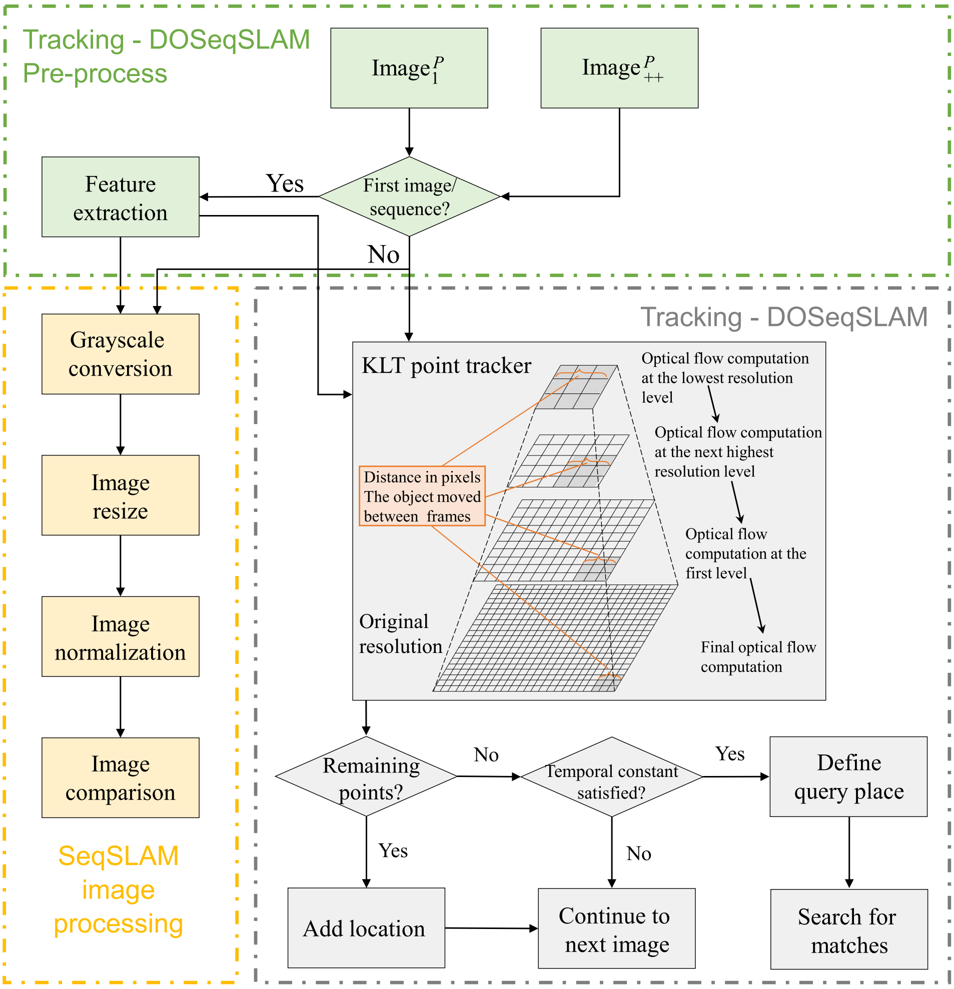


Figure 4.1. *An overview of the proposed sequence-based visual place recognition for simultaneous localization and mapping. As the incoming camera information (I_1^P) arrives to the pipeline, key-points are extracted through the detection and description algorithm of speeded-up robust features (SURF) [86] each time a new sub-map begins its generation. Subsequently, the visual information follows the SeqSLAM's [110] processing steps. In particular, it is downsampled and normalized before compared to the previously visited locations. When the subsequent image (I_{++}^P) enters the system, points are tracked through the Kanade-Lucas-Tomasi (KLT) method [257] to define places dynamically. Finally, when point tracking is lost, and a temporal constant is satisfied, the database is queried with the latest formulated place.*

visited via SAD, and finally, when a temporal constant is satisfied, the database is searched for candidate loop-closures. An outline of the proposed workflow is shown in Fig 4.1.

4.1.1 Efficient places' definition through point tracking

Algorithm 1 Place definition

```
Input: $I$: Incoming image, $P$: Place index, $L^P$: Place length
Output: $P$: Place index, $L^P$: Place length
if $L^P == 0$ then
    SP$_I$ = detectSURF($I$) // extract $\nu$ SURF key-points from $I$
    TP$_{I-1}$ = KLT(SP$_I$) // initialize tracked points
    $numTrackedPoints$ = sum(TP$_{I-1}$)
    $L^P$++
else
    TP$_{I-1}$ = KLT(TP$_I$, I) // track points in $I$
    TP$_{I-1}$ = TP$_I$ // set tracked points for next iteration
    $numTrackedPoints$ = sum(TP$_{I^P}$)
    $L^P$++
end
if $numTrackedPoints < 1$ then
    $P$++
    $L^P$ = 0
end
```

Point tracking is essential for several high-level computer vision tasks, such as motion estimation [258], structure from motion [259], and image registration [260]. Since the earliest works, point trackers have been used as a *de facto* tool for handling points in a video. We chose to adopt a tracker based on a floating-point, local feature detection and description algorithm throughout the navigation procedure. More specifically, in the proposed framework, the sub-maps' are defined owed to a repeatability check of points' occurrence between consecutive image frames. A set of ν key-points detected via SURF (SP$_{I_1^P}$ = $\{$sp$^1_{I_1^P}$, sp$^2_{I_1^P}$,..., sp$^{\nu}_{I_1^P}\}$) in the first location of each place (I_1^P). Next, the points are fed into a KLT tracker along with the subsequent perceived visual measurement (I^P_{++}), yielding a set of tracked points (TP$_{I^P_{++}}$ = $\{$tp$^1_{I^P_{++}}$, tp$^2_{I^P_{++}}$,..., tp$^{\nu}_{I^P_{++}}\}$). Points in I^P_{1++} are browsed within 3 levels of resolution, around a 31×31 patch allowing the system to handle large displacements between images. In such a way, robust sub-maps are generated, even if occlusions occur owing to moving objects, as evidenced by the experimental evaluation in Section 4.2. Furthermore, aiming at a low-complexity pipeline, the computation of bidirectional error between points is avoided. As the algorithm progresses over time, points tend to be lost gradually due to lighting variation or out-of-plane rotation. At time t, when every point's repeatability expires, the previous visual sensory stream $I_{(t-n)}$, ..., $I_{(t-3)}$, $I_{(t-2)}$, $I_{(t-1)}$ is determined as a new place:

$$\left| \bigcap_{i=n}^{i=0} \mathrm{TP}_{I^P_{(t-i)}} \right| \leq 1. \tag{4.1}$$

Finally, two important components are retained during navigation: i) the place index P and ii) its length L^P. Algorithm 1 summarizes this process.

4.1.2 Images' modulation

Afterward, the SeqSLAM's *feature extraction* process follows. In particular, for each image I entering the system, the visual data are converted into the grayscale equivalent and then are downsampled into χ pixels. Next, the resized image is normalized in an N-size local neighborhood and comparisons against the traversed path are performed employing SAD:

$$D_{ij} = \frac{1}{R_x R_y} \sum_{x=0}^{R_x} \sum_{y=0}^{R_y} |\rho^i_{x,y} - \rho^j_{x,y}|, \tag{4.2}$$

where R_x and R_y denote the reduced dimensions of the images, while ρ represents each pixel's intensity value. A vector D_i for location i containing distance metric against every previously visited location j is generated, resulting in comparison matrix D.

4.1.3 Place-to-place association

The query procedure starts when the latest group-of-images is determined. To perform reliable searching for similar sub-maps, the newly generated place P_Q should not share any common semantic information with the recently visited locations. To prevent the proposed pipeline from detecting such cases, we consider a temporal window t_W, which rejects locations visited just earlier (I_1^Q,..., I_1^Q - t_W). This window is defined based on a temporal constant ψ and the place's length L^Q:

$$t_W = \psi + L^Q. \tag{4.3}$$

This way, the searching area spans among the first perceived location I_1 and the one determined by the temporal window $I_1^Q - t_W$ as depicted in Fig. 4.2 by the red dashed line. The latest produced sub-map seeks into the navigated path for similar places via a sequence-based technique. For each database location I_j, belonging to the searching area, a difference score s is calculated by averaging the accumulated values:

$$s_j = \frac{1}{L^Q} \sum_{I_1^Q}^{I_{end}^Q} D_{jk}. \tag{4.4}$$

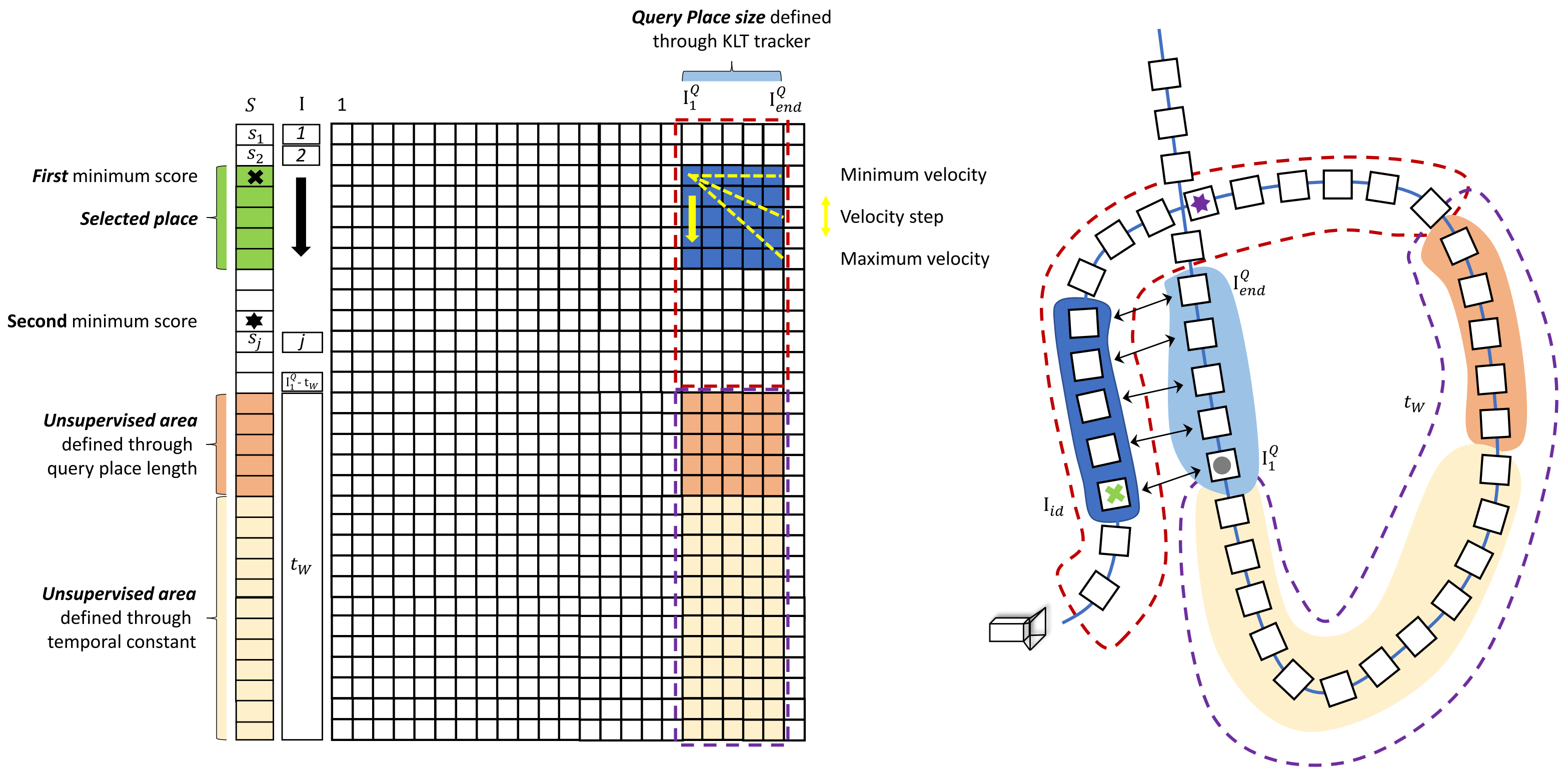


Figure 4.2. *A representation of the proposed framework. To define sub-maps, local key-points are extracted via the speeded-up robust features (SURF) [86] detector for the first image of each place (gray circle). Through Kanade-Lucas-Tomasi (KLT) method [257], points are tracked along the navigated path, while a new group-of-images is determined when each tracked point is lost. The query process begins when a temporal window based on a time constant and query length is satisfied (beige and light orange area). The latest generated sub-map (light blue area) seeks for similar places along the traversed route in a sequence-based scheme. Each visited location (j) is associated with a score (s_j) which indicates the nearest neighboring trajectory assumption (yellow dashed line). The selected images (green box in S vector) point out the proper place and an image-to-image search is subsequently performed.*

Algorithm 2 Detecting loop places

```
Input: $D$: Difference matrix, $P$: Query place index, $L$: Query place length, $f$: dataset's
       frame rate
Output: $id$: Candidate index, $score$: Candidate score
$t_W$ = 40 * $f$ + $L$ // temporal window definition
for each image $I_j$ in Database do
    T = computeTrajectoryScores($I_j$, $D$, $P$, $L$)
    t = min(T)
    S($I_j$) = t
end
[$id$, $score1$] = min(S) // find the minimum score and candidate index
$e$ = [$I_{id-L/2}$, ..., $I_{id+L/2}$] // define images around $I_{id}$
S($e$) = $\infty \setminus \setminus$ reject images in $e$
[$\sim$, $score2$] = min(S) // find the second minimum score excluding images in $e$
$score$ = $score1/score2$ // compute the normalized score for $I_{id}$
```

In the above, I_1^Q and I_{end}^Q are the first and last image-timestamps of the query sub-map, respectively, L^Q is the query's length and k denotes velocity assumption paths:

$$k = j + V(L^Q - I + t), \tag{4.5}$$

where V is designated by multiple values within the range of $[V_{min}, V_{max}]$ (advancing by V_{step} each time step t).

These scores are based on the values the trajectory line passes through in travelling from I_1^Q to I_{end}^Q (see Fig. 4.2). The trajectory with the minimum s value is selected as the representing score s_j between the query place and the one starting from image frame I_j. When all database images have been examined, a score vector is determined $S = \{s_1, s_2, ..., s_{I_1^Q - t_W}\}$ and subsequently the minimum value is selected corresponding to the start location I_{id} of the candidate place. Next, following the *nearest neighbor distance ratio* [112] the chosen score is normalized over the second lowest score outside of a window range of equal size with the place's length L^Q. The normalized score, which is the ratio between these scores, is calculated for each place, while one of the following conditions have to be satisfied before a sub-map is recognized as loop-closure candidate. The recent score has to be lower than a threshold th (th < 0.7 [110]) or the score generated by the last two consecutive sub-maps to satisfy a threshold θ. This temporal consistency check is incorporated in the proposed pipeline since loop-closure detection is a task submitting to a temporal order of the visited areas throughout the navigation. That is, if a place is identified as known, then it is highly probable that the following ones have also gone through. This way, we achieve to improve the system's performance, while the system avoids to lose actual loop detections due to strict thresholding. Algorithm 2 illustrates this process.

4.1.4 Local best match

Up to this point, the proposed algorithm is capable of identifying a previously visited place in the navigated map. Finally, an image-to-image correlation is performed between the query's locations and the most similar members of the selected sub-map in the database. Hence, each place's member is associated to the most similar from the corresponding ones in the matched database image through the SAD sub-matrix. Let us consider that at time t, the system correctly indicates a previously visited place by matching pair $\langle I_1^Q, I_{id} \rangle$. Our method defines a group-of-images which are the only set of database entries that are going to be evaluated through SAD metrics. We determine this group to be of double the size of cameras frequency κ, while is centered around I_{id} for I_1^Q, i.e., $I_{(id-1)-\kappa}$,..., $I_{(id-1)+\kappa}$. Hence, for the following image in the query place I_2^Q this area shifts by one.

4.2 Experimental results

Table 4.1. *Parameters utilized from the proposed sequence-based pipeline. Most of the reported values come from the OpenSeqSLAM implementation, while the rest are selected by means of our experiments.*

Parameter	Symbol	Value
Downsampled image size	χ	2048
Patch normalisation length	N	8
Reduced image size	R_x, R_y	32, 64
Minimum velocity	V_{min}	0.8
Velocity step	V_{step}	0.1
Maximum velocity	V_{max}	1.2
Extracted SURF points	ν	500
Search area time constant	ψ	40 sec

This section provides an extensive evaluation of the proposed pipeline and its comparative results. Precision-recall metrics and the ground truth information are utilized to assess the algorithm performance in a total of seven datasets. Subsequently, the presented approach is compared with the baseline version of SeqSLAM, as well as other modern place recognition solutions. Moreover, aiming to exhibit the system's low-complexity, comparisons are performed against a modified version of SeqSLAM, dubbed as “DOSeqSLAM" (dynamic and online sequence-based place recognition for SLAM). Comparisons are performed based on the parameters in Table 4.1. Those values remain constant for every tested environment in order to prove the adaptability of Tracking-DOSeqSLAM.

4.2.1 DOSeqSLAM

To define a dynamic sub-map in DOSeqSLAM, the process presented in Chapter 3 is followed. First, local key-points are detected via SURF from each incoming visual sensory data, and through a features' matching coherence check, new places are determined throughout the robot's traversed path. To identify a previously visited location, the searching process is based on a similar procedure as the one used in Tracking-DOSeqSLAM, i.e., when the latest place P_Q has defined, comparisons with the database are performed for the first frame in the previous generated sub-map P_{Q-1}. Next, several trajectories are projected on the distance matrix D for every traversed location j, and, subsequently, multiple scores s are calculated corresponding to different trajectory assumptions (Equation 4.5) by averaging the accumulated values (Equation 4.4). Afterwards, the minimum score s is selected, yielding an S vector, wherein the lowest value is chosen for the particular location I_j. This score is normalized over the second lowest value outside of a window:

$$W = P^{Q-2} * 2 + P^{Q} * 2 \tag{4.6}$$

resulting to γ. At last, an average weighted filter is applied for the final decision:

$$\mathbf{f}(\gamma) = \frac{1}{6}\gamma_{(P^{Q-2})} + \frac{4}{6}\gamma_{(P^{Q-1})} + \frac{1}{6}\gamma_{(P^{Q})}. \tag{4.7}$$

A candidate loop-closure place is accepted when factor γ is satisfied. At last, the system performs a greedy image-to-image search into the SAD sub-matrix for single image associations.

4.2.2 Parameters' discussion

In this subsection, we briefly discuss the system's chosen parameters. In general, most of the proposed values, e.g., downsampled image size χ, image's reduced size R_x, R_y, come from the initial version of SeqSLAM. Velocity's properties $[V_{max}, V_{min}, V_{step}]$ and the normalization parameter N are defined based on the open-source implementation of OpenSeqSLAM2.0 Matlab toolbox [236]. Extracted key-points ν are defined via the precision-recall metrics in Fig. 4.3.

4.2.3 Performance evaluation

By altering the loop-closure decision parameter θ, precision-recall curves are monitored for different cases of image's key-points detection (ν = 100, 300, 500, 700) in Fig. 4.3. The system's performance for the proposed dynamic sub-maps' generation is evaluated and compared against the approach of SeqSLAM and DOSeqSLAM. Furthermore, a 40 sec temporal window, similar to the proposed method, is applied to both solutions in

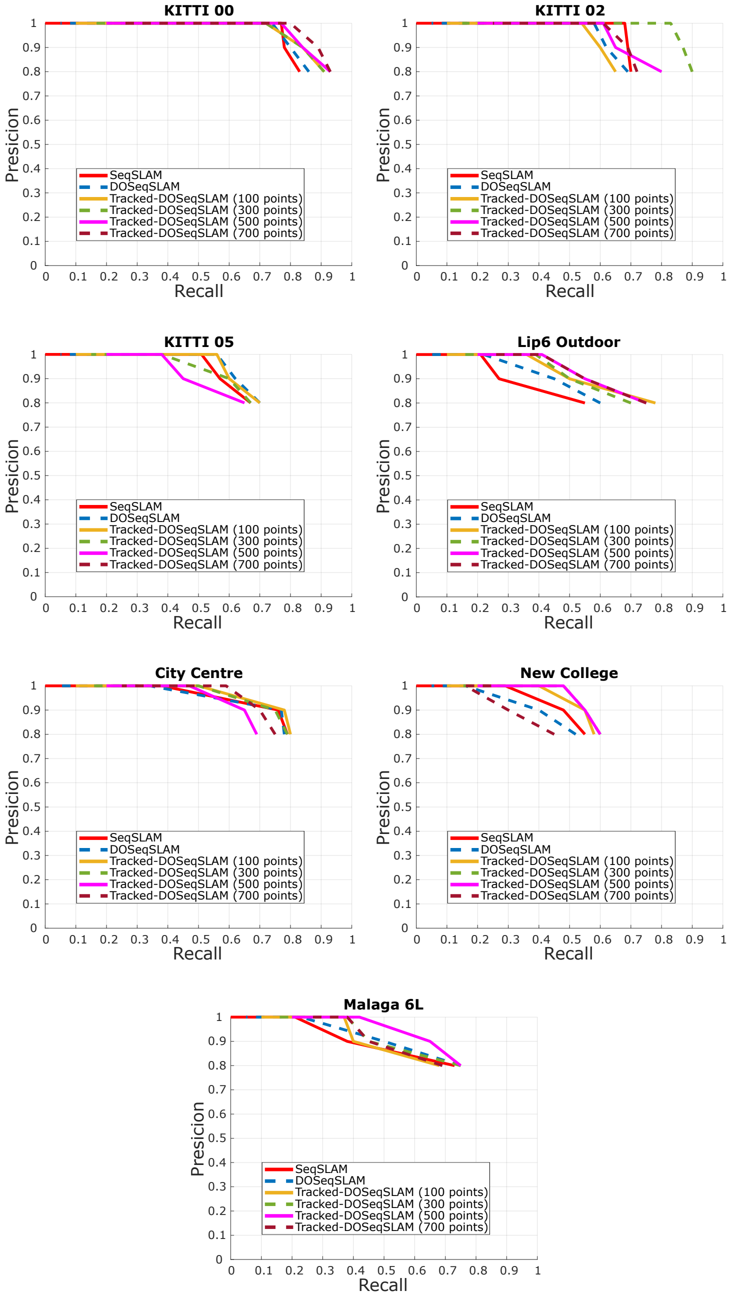


Figure 4.3. *Precision-recall curves evaluating the utilized number ν of extracted key-points of speeded-up robust features (SURF) [86] against SeqSLAM and DOSeqSLAM. Experiments are performed on KITTI courses 00, 02, and 05 [250], Lip 6 Outdoor [140], Oxford City Centre [129], New College vision suite [254], and Malaga parking 6L [252]. As the number of detected key-points increases, the proposed system presents a slight improvement, reaching recall values of about 77% in case of KITTI course 00, 85% in KITTI course 02, and 56% in KITTI course 05. In Lip 6 outdoor, a score of 50% is achieved, while a similar performance is observer for the rest datasets. However, the performance falls drastically when key-points' detection exceeds the amount of 500, as evidenced in KITTI course 05 and New College. This is mainly owed to the resulting size of the generated sub-maps which fail to be matched with the ones in the traversed trajectory.*

Table 4.2. *Recall rates at 100% precision (R_{P100}): a comparison of the proposed method against SeqSLAM and DOSeqSLAM. Bold values indicate the maximum performance per evaluated dataset. As shown from the obtained results, Tracking-DOSeqSLAM outperforms the other methods. A performance improvement is observed as the extracted set of key-points increases until a certain point. Aiming for an efficient system which preserves high recall scores for 100% precision, the case of 500 points is indicated.*

Dataset	SeqSLAM	DOSeqSLAM	Tracking-DOSeqSLAM (100 points)	Tracking-DOSeqSLAM (300 points)	Tracking-DOSeqSLAM (500 points)	Tracking-DOSeqSLAM (700 points)
KITTI course 00	77.3	74.8	69.8	72.1	**77.6**	80.1
KITTI course 02	68.2	58.9	54.6	**83.6**	61.1	61.1
KITTI course 05	51.5	56.7	**56.7**	38.4	38.2	–
Lip 6 outdoor	21.2	22.5	36.8	39.8	**40.9**	40.9
Oxford City Centre	38.9	34.9	50.4	50.4	47.1	**59.4**
New College	29.3	16.8	39.9	40.0	**40.0**	16.3
Malaga parking 6L	21.5	23.3	37.2	38.4	**42.0**	38.1

order to reject early visited locations. For an easier understanding of the curves, the best results at 100% precision (R_{P100}) are also given in Table 4.2. Our first remark is that the area under the curve of Tracking-DOSeqSLAM is higher than the corresponding curves for the other pipelines, outperforming them in most of the evaluated datasets. As can be observed, DOSeqSLAM can obtain similar recall at perfect precision as SeqSLAM, except for New College, where the result drops to a rate of 17%. According to our experiments, the proposed pipeline shows exceptionally high performance for Lip 6 outdoor, City Centre, and Malaga parking 6L, compared to the other solutions for each case of the extracted key-points. In addition, the maximum scores for the other datasets are also high, while a significant improvement is observed in KITTI course 02 for a total of 300 key-points, reaching a score of about 85% for perfect precision.

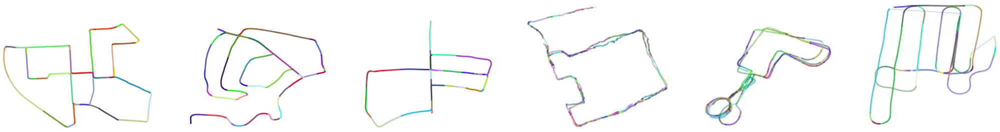


Figure 4.4. *Sub-maps generated from the proposed method. Parameters are defined as presented in Table 4.1. Images exhibiting time and content proximity are labeled by the same color. From left to right, sub-maps are illustrated for KITTI [250] courses 00, 02, 05, Oxford City Centre [129], New College [254], and Malaga parking 6L [252]. 47, 52, 22, 151, 119, and 43 places are generated, respectively. As can be seen in most of the cases, the images are tagged with the same color when the robot traverses a route which presents similar visual content. This is especially highlighted in the KITTI image-sequences, where the camera measurements arrive from a forward moving car, in contract to City Centre's lateral camera orientation.*

However, counter to most datasets, where the increased key-points' extraction improves the performance, evaluating the proposed method in KITTI course 05 and New College shows an instant drop in the recall rate. In the latter case, we observe a lower score, while in the former one, the proposed method is unable to recognize previously visited areas. This is because places generated under those conditions fail to match with the database due to their extreme size. By considering the results presented in Table 4.2, the parameter ν is selected at 500 with the aim to ensure a system that achieves high recall scores for 100% precision. Fig. 4.4 shows the sub-maps formulated

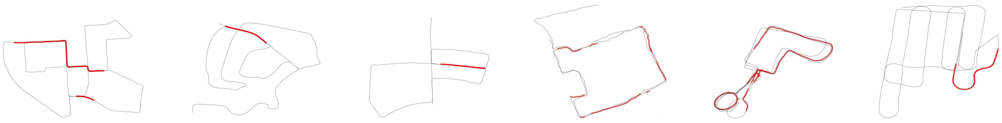


Figure 4.5. *Loop-closures detected by the proposed pipeline for each dataset. From left to right: KITTI [250] courses 00, 02, 05, Oxford City Centre [129], New College [254], and Malaga parking 6L [252]. Red dots indicate that the system closes a loop with another image in the database.*

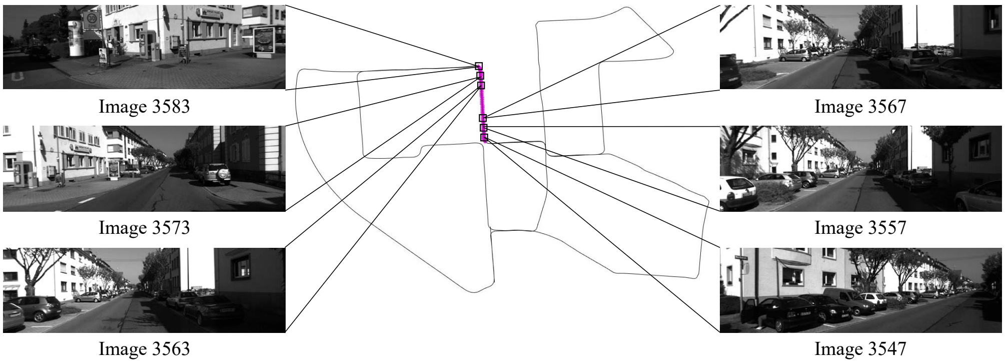


Figure 4.6. *An illustrative example of our place generation technique based on point tracking. The respective camera poses corresponding to the same group of images are marked in magenta. A set of key-points extracted via speeded-up robust features (SURF) [86] is detected in the first location of a newly formulated place (image 3547) and subsequently tracked along the trajectory. At time t (image 3583), the incoming visual sensory stream $I_{(t-n)}$, ..., $I_{(2)}$, $I_{(1)}$, $I_{(t)}$, is finalized as a new place since all the initial points cease to exist from the tracker.*

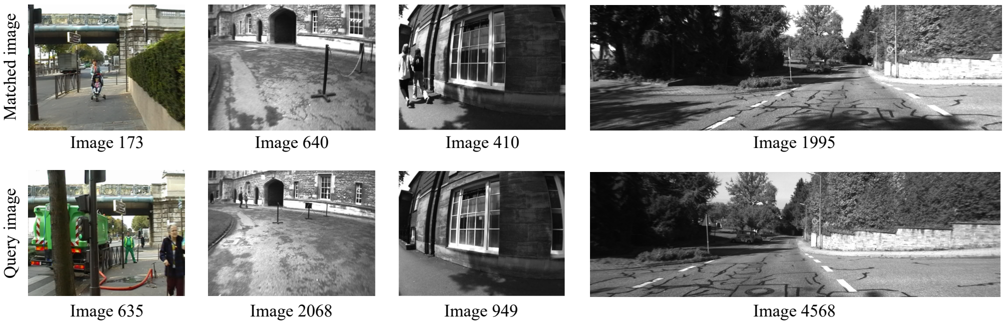


Figure 4.7. *Some example images that are correctly recognized by our pipeline as loop-closure events. The query frame is the image recorded by the vehicle at time t, whereas the matched image frame is the corresponding one identified among the members of the chosen place. From left to right: Lip 6 outdoor [140], New College [254], Oxford City Centre [129], and KITTI course 02 [250].*

by Tracking-DOSeqSLAM for each dataset, while Fig. 4.5 presents the detected loops. A random color has been assigned to highlight a different place across the traversed trajectory for each sub-map. Thus, every location associated with the same place is labeled by the same color. An example containing images from the same group-of-images defined by our algorithm's point tracking is illustrated in Fig. 4.6. Evidently, as soon as the robot turns to a visually consistent route, the corresponding images that exhibit time and content proximity are aggregated in the same group. Finally, in Fig. 4.7, some accurately detected locations using the selected parameterization are shown.

4.2.4 System's response

Table 4.3. *Processing time per image (ms/query) of Tracking-DOSeqSLAM, DOSeqSLAM, and SeqSLAM, for KITTI course 00 [250]. It is notable that the proposed pipeline requires less time due to its efficient matching process, which is based on the image aggregation from the generated places.*

		Average time (ms)		
		SeqSLAM	DOSeqSLAM	Tracking-DOSeqSLAM
Feature extraction	Key-point detection	–	42.67	0.01
	Key-point description	–	27.96	–
	Resize	2.43	2.43	2.43
	Patch normalization	5.77	5.77	5.77
Environment representation	Key-point tracking	–	–	4.27
	Feature matching	–	6.75	–
Decision making	Comparison (SAD)	42.90	42.90	42.90
	Matching	67.66	64.18	0.71
Total pipeline		118.76	192.66	56.20

To analyze the computational complexity of the proposed method, we ran each framework, i.e., SeqSLAM, DOSeqSLAM, and Tracking-DOSeqSLAM, on KITTI course 00, which is the longest among the evaluated ones exhibiting a remarkable amount of loop-closures. In Table 4.3, an extensive assessment of the corresponding response time per image is presented. The detection and description of SURF key-points, the image resize, and the patch normalization constitute the *feature extraction* process presented every evaluated method. Key-point tracking corresponds to the time needed by the KLT tracker, while feature matching the time for DOSeqSLAM to segment the incoming visual stream. The *decision making* module is the time needed for the images' comparison through SAD. Lastly, matching denotes the timing for each method to search for similar places in the database. The results show that the proposed system can reliably detect loops, while maintaining very low execution times. It is observed that every involved step is notably fast except for the comparison process which exhibits the highest execution owed to the utilized metric technique. The time for key-point extraction is negligible as we search for new elements at the beginning of a new place, while the timing for point tracking is also low.

4.2.5 Comparative results

This section compares the proposed pipeline against other algorithms concerning the complexity and performance. Aiming to exhibit the low-complexity of the proposed

Table 4.4. *In depth comparison with the baseline versions of the proposed method. As can be observed from the average computational times (T) and number of generated places, Tracking-DOSeqSLAM achieves substantially lower timings for each evaluated case, outperforming the rest solutions.*

	SeqSLAM		DOSeqSLAM		Tracking-DOSeqSLAM	
Dataset	Places	T(ms)	Places	T(ms)	Places	T(ms)
KITTI course 00	4554	118.76	155	192.66	**47**	**56.20**
KITTI course 02	4661	123.04	378	209.02	**52**	**58.99**
KITTI course 05	2761	58.62	192	132.90	**22**	**38.66**
Lip 6 Outdoor	1063	22.13	177	37.56	**51**	**20.87**
Oxford City Centre	1237	26.96	211	71.41	**151**	**25.47**
New College	2624	55.51	323	94.70	**119**	**36.28**
Malaga parking 6L	3474	81.73	162	186.22	**43**	**48.97**

system, we present in Table 4.4 the final amount of generated places and the average processing time for Tracking-DOSeqSLAM and the baseline versions, viz., DOSeqSLAM and SeqSLAM. In this regard, we show that the proposed modifications result in a improvement in terms of processing time and computational complexity. Since the proposed method adopts the same *feature extraction* module, e.g., image downsample, and *decision making* process, e.g., SAD comparisons, the computational complexity mainly depends on the number of constructed places. As highlighted in Table 4.4, our system achieves the generation of an amount of places at least one order of magnitude less than SeqSLAM, while a significant decrease is also presented against DOSeqSLAM. This results to notably fast associations between similar sub-maps permitting the proposed framework to process in lessen time in contrast to the other versions, while presenting high recall scores for perfect precision as evident in most of the tested datasets.

Furthermore, with the aim to help the reader to identify the contribution of the proposed pipeline, as well as for the sake of completeness, in Table 4.5, we show the results against the two modern approaches presented in Chapter 3. Albeit the proposed system achieves high recall rates in every tested dataset, the difficulty to present higher scores against the methods which utilize more sophisticated *feature extraction* modules for the locations' representation is evident. This is owed to the inability of SAD to quantify the obtained image visual properties. However, our main purpose is to demonstrate the achieved performance gain, over the baseline versions, through a refined trajectory segmentation, while operating with the lowest possible complexity and avoiding any training procedure. Thus, a direct comparison of Tracking-DOSeqSLAM with the rest of the approaches is not informative; it is only included here as a performance indicator to better interpret the possible improvement margins. On the support of thereof, Table 4.5 compares the average execution time of each method on

Table 4.5. *Performance comparison with other appearance-based place recognition methods. Recall scores for 100% precision and average computational times are given.*

	iBoW-LCD		iGNGmap-LCD		Tracking-DOSeqSLAM	
Dataset	R(%)	T(ms)	R(%)	T(ms)	R(%)	T(ms)
KITTI course 00	76.5	400.2	**93.1**	527.6	77.6	**56.20**
KITTI course 02	72.2	422.3	**76.0**	553.3	61.1	**58.99**
KITTI course 05	53.0	366.5	**94.2**	402.8	38.2	**38.66**
Lip 6 outdoor	**85.2**	228.0	12.0	198.5	40.9	**20.87**
City Centre	**88.2**	336.2	16.3	232.7	47.1	**25.47**
New College	73.1	383.7	**88.0**	302.1	40.0	**36.28**
Malaga parking 6L	57.4	440.8	**87.9**	553.6	42.0	**48.97**

the representative datasets. It is noteworthy that the proposed pipeline can achieve the lowest timings in every tested image-sequence reaching one order of magnitude lower times against the other solutions.

5 Modest-vocabulary loop-closure detection with incremental bag of tracked words

As our our interest lies in developing a low-complexity and effectiveness appearance-based place recognition framework that identifies loop-closures, in this chapter we propose an incremental BoW method based on feature tracking, that is, the Bag of Tracked Words (BoTW). Exploiting the advantages presented in the previous chapters, the KLT point tracker is utilized to formulate such a vocabulary, accompanied by a guided feature selection technique. Each point whose track ceases to exist is transformed into a visual word, viz., tracked word, used to describe every key-point element contributing to its formulation. The query's tracked descriptors seek for the nearest neighboring words into the vocabulary to detect loop-closures, distributing votes across the traversed path. Each voted location is assigned with a similarity score through a binomial probability density function, which is meant to indicate candidate matches.

In addition, we further introduce modeling mechanisms that significantly improve our framework's memory usage and computational complexity. The unchecked generation of new elements in incremental vocabulary methods affects the systems' performance since these new entries reduce their distinctive ability, especially in cases where the robot traverses pre-visited locations. To this end, the proposed method applies a map management scheme that restricts similar words during the vocabulary construction to address such a deficiency. Moreover, loop-closure detection is a task submitting to a temporal order of the visited areas along the navigation route. If a location is identified as previously visited, then it is highly probable that the following ones have also gone through. This property is explored in the proposed approach by employing a Bayes filter, accompanied by a temporal consistency constraint, over the probabilistic scores produced through the binomial probability density function. We specifically exploit the temporal information of the incoming visual stream to decide about the appropriate belief state. This way, an improvement in recall rate is achieved since locations within a known area are not excluded even if they present a lower similarity than the defined threshold. Lastly, a geometrical verification step is performed over the most similar candidates. The proposed method is tested on nine different environments in a broad set of conditions and compared against various methods. In the following section, our approach's implementation blocks are discussed in detail, while its evaluation is presented in Section 5.2.

5.1 Methodology

The workflow to carry out the place recognition task is comprised of two parts: i) the bag of tracked words (BoTW), and ii) the probabilistic loop-closure detection. Regarding the first, it includes the components needed for the database generation, while the second part the ones for searching and recognizing previously visited locations. The following sections describe the individual parts of the algorithm in detail.

5.1.1 Bag of tracked words

Similar to the approaches mentioned in the previous chapters, the proposed one does not require any training process or environment-specific parameter tuning since the map is built on-line in the course of the robot's navigation. As the construction of the vocabulary plays the primary role in the proposed appearance-based loop-closure detection pipeline, it has to be as discriminable and detailed as possible. Our trajectory mapping is based on the observation that the traversed path is associated with unique visual words generated incrementally. On the contrary, through the BoTW scheme, each codeword is initiated by a local key-point tracked along the trajectory in consecutive camera frames. An algorithm with scale- and rotation-invariant properties has been adopted to obtain a robust and accurate description against image deformations. Overall, the map representation during the robot's navigation consists of four individual parts: i) feature tracking, ii) guided feature selection, iii) tracked word generation, and iv) merging words.

5.1.1.1 Feature tracking

We have chosen to map the trajectory, through a tracker based on SURF. Each extracted element has a detection response that quantifies its distinctiveness among the rest of the image's content. This property is used to select the most prominent local key-points in the image. Thus, intending to promote computational efficiency, we limit the number of features to be used to the ν most prominent. Those key-points ($\mathrm{P}_{t-1} = \{\mathrm{p}^1_{t-1}, \mathrm{p}^2_{t-1},..., \mathrm{p}^{\nu}_{t-1}\}$) from the previous image I_{t-1}, along with the current camera frame I_t, are utilized within a KLT point tracker, to obtain their projected location, which we refer to as tracked points ($\mathrm{TP}_t = \{\mathrm{tp}^1_t, \mathrm{tp}^2_t,..., \mathrm{tp}^{\nu}_t\}$). Additionally, we retain the corresponding set of description vectors ($\mathrm{D}_{t-1} = \{\mathrm{d}^1_{t-1}, \mathrm{d}^2_{t-1},..., \mathrm{d}^{\nu}_{t-1}\}$) that are meant to be matched with the corresponding ones (D_t) in I_t.

5.1.1.2 Guided feature selection

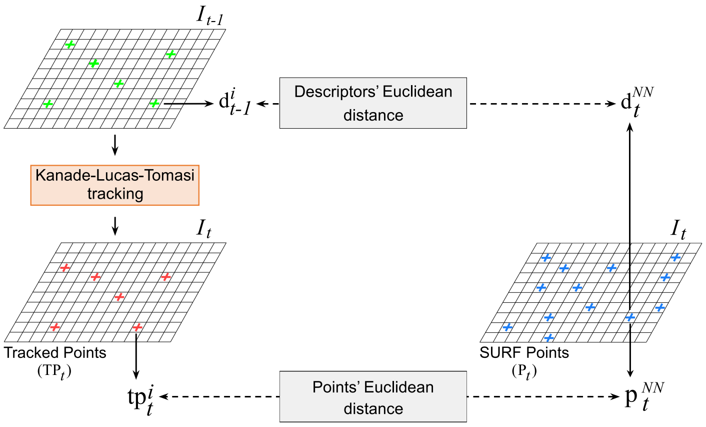


Figure 5.1. *Guided feature selection over the points being tracked. The Kanade-Lucas-Tomasi [257] tracker estimates the expected coordinates for each of the Tracked Points (TP_t = $\{tp^1_t, tp^2_t,..., tp^\nu_t\}$), originated from the previous image I_{t-1}, to the current one I_t, (the green and red crosses (+), respectively). Their nearest-neighboring points $p^{NN}_t \in P_t$, detected via speeded-up robust features [86], are evaluated as per their points' coordinates and descriptors distance for the proper feature selection using equations 5.1 and 5.2.*

Although KLT is sufficiently effective in estimating a detected point's flow between successive frames (e.g., I_{t-1} and I_t), accumulative errors within the entire image-sequence may drift the tracked points. Points have to be periodically redetermined to track features over a long period. Having apprehended these challenges, we adopt a guided feature selection technique [261] (Fig. 5.1) that, along with the KLT's flow estimation, also detects new SURF key-points (P_t = $\{\mathrm{p}^1_t, \mathrm{p}^2_t,... \mathrm{p}^\mu_t\}$) and computes the corresponding description vectors (D_t = $\{\mathrm{d}^1_t, \mathrm{d}^2_t,... \mathrm{d}^\mu_t\}$) from the most recent frame I_t. Note that we retain only the μ most prominent detected feature points with a response higher than Φ, to reduce computational complexity further. A k-NN (k = 1) search is performed between the tracked points' coordinate space (TP_t) detected in image I_t and the ones in P_t. Thus, for each tracked point tp^i_t, the nearest $\mathrm{p}^{\mathrm{NN}}_t \in \mathrm{P}_t$ is accepted as a proper extension-member of the track, providing that the following conditions are satisfied:

– the Euclidean distance between tp^i_t and its corresponding $\mathrm{p}^{\mathrm{NN}}_t$ is lower than α:

$$\ell_2(\mathrm{tp}^i_t, \mathrm{p}^{\mathrm{NN}}_t) < \alpha, \tag{5.1}$$

– the Euclidean distance between its descriptor $\mathrm{d}^{\mathrm{NN}}_t$ and the d^i_{t-1}, corresponding to p^i_{t-1} in the previous image I_{t-1}, is lower than β:

$$\ell_2(\mathrm{d}^i_{t-1}, \mathrm{d}^{\mathrm{NN}}_t) < \beta. \tag{5.2}$$

If at least one of the above conditions is not met, the corresponding track point ceases to exist, and it is replaced by a new one detected in I_t, ensuring a constant number of ν TP_t members. Similarly, aiming to preserve a constant set of points during the robot's navigation, when a tracked feature is discontinued (regardless of whether it forms a tracked word or not), it is replaced by a new one, fished out from I_t. This way, the computationally costly brute force local features' matching as tracking scheme is avoided, while a robust trajectory mapping is achieved.

5.1.1.3 Tracked word generation

The next step of the BoTW procedure is the descriptors' merging, which, in turn leads to the formulation of the visual codewords. When the tracking of a certain point terminates, its total length τ, measured in consecutive image frames, determines whether a new word should be created ($\tau > \rho$). Describing part of the environment, the representative tracked word is computed as the median of the tracked descriptors:

$$\widetilde{\mathrm{TW}}[i] = \mathrm{median}(\mathrm{d}_1[i], \mathrm{d}_2[i], ..., \mathrm{d}_j[i]), \tag{5.3}$$

where $\mathrm{d}_j[i]$ denotes the element in the i-th (SURF: $i \in [1, 64]$) dimension of the j-th ($j \in [1, \tau]$) description vector. Note that, we refer to the tracked word set as a visual vocabulary since each codeword is created through an average representation, which is also the norm for a typical BoW representation. In general, new codewords are generated through averaging the corresponding descriptors, yet in the proposed approach, the median is selected since it provides better performance with lower computational cost as evidenced by the experimental evaluation in Section 5.2. Finally, an indexing list *Idx* is retained that includes the locations upon which each word is tracked in the trajectory.

5.1.1.4 Merging words

Finally, to provide a discriminative visual vocabulary, we avoid adding new visual elements into the vocabulary without comparing their similarity to the database. In the proposed system, an additional preliminary step is incorporated. For each newly generated element, a one-vs-all scheme computes the pairwise distances against the database's ones. Subsequently, the *nearest-neighbor distance ratio* [112] is applied, indicating two visual elements as similar when a distance ratio value lower than 0.5 is satisfied. The tracked descriptors of the newly created element and the vocabulary's chosen word are merged based on equation 5.3, and the new codeword is ignored. However, in Section 5.1.2.6, we further propose a vocabulary management scheme in which visual words corresponding to already visited locations are discarded, resulting in an overall reduced memory footprint.

5.1.2 Probabilistic loop-closure detection pipeline

In this section, our probabilistic framework for the identification of loops within BoTW-LCD is presented. The voting procedure is being described as the first step of the proposed on-line pipeline. Subsequently, we show how the locations are assigned with a probabilistic score through the binomial probability density function, while the derivation of the Bayes filtering scheme used for the estimation of the loop-closure state is also detailed. Finally, we focus on the additional implementation details we adopted for incorporating geometrical verification and visual vocabulary management.

5.1.2.1 Searching the database

With the aim to perform reliable searching during query, the newly acquired frame I_Q should not share any common features with recently visited locations. To prevent our pipeline from detecting cases of early visited locations, we consider a temporal window w, which rejects locations visited just earlier $(I_{Q-1}, I_{Q-2}, ..., I_{Q-w})$. We define this window as $w = t - 4c$, where c corresponds to the length of the longest active point track, as indicated by the retained τ values. In this way, it is guaranteed that I_Q will not share any visual information with the recently created database entries, while at the same time, we avoid the use of a fixed timing threshold that is typically selected by environment-specific experimentation.

Due to the lack of global descriptors for image representation, the proposed appearance-based framework adopts a probabilistic voting scheme to infer previously visited locations. At query time, the most recent incoming sensory data I_Q directly distributes its descriptors –formulated by guided feature selection– to the database via a k-NN (k=1) search among the available database tracked words in a brute force manner. In order to accelerate the matching process, many approaches build a k-d tree [262]. While offering an increased computational performance when applied to a low dimensional descriptor space, the tree is unsuitable for on-line developed vocabularies. This is owed to possible unbalanced branches and the addition of new descriptors after the tree construction, impairing the performance [124]. Moreover, during on-line navigation, the complexity concerning the tree building will eventually prevent real-time processing, especially in large-scale environments containing thousands of images [263]. Besides, our descriptor has a 64-dimensional feature vector, and the k-d tree is unable to provide speedup over the exhaustive search for more than about 10-dimensional spaces [112]. A valid alternative for high dimensional descriptors, as well as for larger vocabularies, is the inverted multi-index file system [264]. This technique is multiple times faster compared to a k-d tree while offering similar performance. However, it needs to be trained beforehand, impractical for incremental approaches within a SLAM framework. Aiming to improve the descriptor matching speed, an incremental feature-based tree is proposed by [265], which is still incompatible with our framework due to its boolean structure. Even though the unavailability of indexing approaches, our work (see Section 5.1.2.6) aims to map the environment efficiently. Therefore, we focus on the significant reduction of the vocabulary's size, as well as the rate of its increment, reaching a footprint of one order of magnitude shorter than other state-of-the-art techniques. As our results in Section 5.2.3 suggest, using such a small vocabulary renders the complexity of an exhaustive search inferior to the overhead of retaining a dynamic indexing file system.

5.1.2.2 Navigation using probabilistic scoring

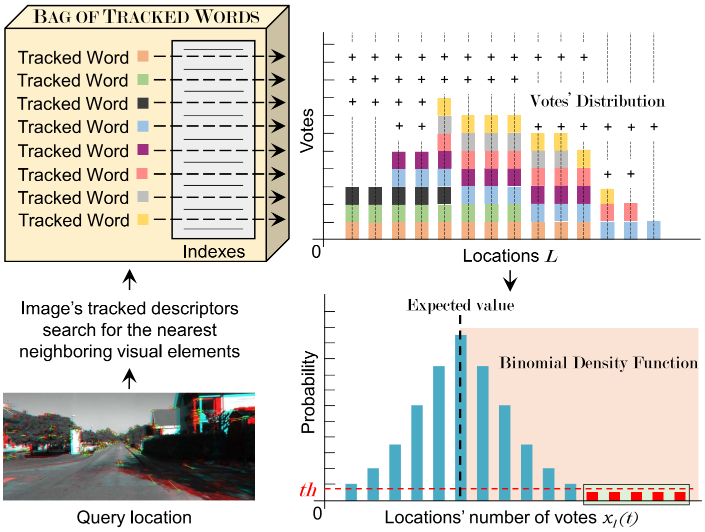


Figure 5.2. *Probabilistic appearance-based loop-closure detection. During a query event, the most recently obtained image directly distributes its descriptors, formulated by guided feature selection, to the bag of tracked words list via a greedy nearest-neighbor search. Votes are assigned to the map L, whilst a vote counter for each location $l \in L$ increases relatively to the contributing words (colored cubes). Finally, candidate locations are indicated via a binomial density function according to their vote density $x_l(t)$. Highlighted with red, instances of votes' count correspond to locations that are intended for a geometrical check since they satisfy the rareness limit th of a loop-closure, while also exceeding the expected vote aggregation value.*

During the matching process among the query features from I_Q and the vocabulary, votes are distributed into the map L under the tracked words' indexing list *Idx*, as depicted in Fig. 5.2. A database vote counter $x_l(t)$ for each traversed location $l \in [1, t-4c]$ increases in agreement with the associated words. To avoid the simplified approach of thresholding the accumulated number of votes, a binomial probability density function is adopted to assign a score over each location based on the votes' density:

$$X_l(t) \sim Bin(n,p), n = N(t), p = \frac{\lambda_l}{\Lambda(t)}, \tag{5.4}$$

where $X_l(t)$ represents the random variable regarding the number of accumulated votes of each database location l at time t, N denotes the multitude of query's tracked words (the cardinality of TP$_Q$ after the guided feature selection), λ is the number of visual elements included in l (the cardinality of TW$_l$) and $\Lambda(t)$ corresponds to the size of the generated BoTW list until t (excluding the locations inside the window w). The binomial expected value of a location l has to satisfy a loop-closure threshold th, so as to be accepted:

$$Pr(X_l(t) = x_l(t)) < th < 1, \tag{5.5}$$

where $x_l(t)$ corresponds to the respective location's aggregated votes. However, to avoid cases where a location accumulates unexpectedly few votes due to extreme dissimilarities, the following condition should also hold:

$$x_l(t) > E[X_l(t)]. \tag{5.6}$$

Conditions 5.5 and 5.6 of binomial probability density function are depicted in Fig. 5.2 through the light green and light orange areas, respectively. In addition, with the aim to avoid the redundant computation of probabilistic scores for each traversed location (e.g., for completely unvoted entries), we propose to compute the binomial-based score only for locations gathering more than 1% of the votes distributed by the tracked descriptors.

5.1.2.3 Location estimation via recursive Bayes rule

The heuristic temporal consistency check proposed in chapter 3, wherein a location is accepted as a loop-closure event when the system meets specific conditions for a certain sequence of consecutive measurements presents the disadvantage that many loop hypothesis belonging at the starting location of a pre-visited area are ignored until the temporal check is satisfied. With a view to tackle this drawback, we take advantage of the temporally consistent acquisition of images within the loop-closure task and adopt a Bayesian scheme. Even though this approach can be considered to be a standard practice in the field [95, 145, 149, 266] our system differs in the aspect that we chose to apply a simple temporal model which maintains the decision factor between consecutive observations, rather than to compute a probability score for each database entry. The discrete Bayes filter allows us to deal with noisy measurements and ensures temporal coherency between consecutive predictions, integrating past estimations over time. Despite the presence of the Bayesian filter, locations captured in a sequence of loop-closing images are processed for further evaluation without being affected by their binomial-based score.

A proper filtering algorithm needs to maintain only the past state's estimates and updating them, rather than going back over the entire history of observations for each update. In other words, given the filtering result up to time $t-1$, the agent needs to compute the posterior (filtering) distribution $p(S_t \mid O_t)$ for t using the new observation

O_t. Let $S_t = \langle No\ Loop, Loop \rangle$ be the state variable representing the event that I_t closes a loop, while O_t is the binomial response $Pr(X_l(t) = x_l(t))$ between I_Q and the database. Following the Bayes' rule and under the Markov assumption, the posterior can be decomposed into:

$$p(S_t|O_t) = \eta \underbrace{p(O_t|S_t)}_{\text{Observation}} \underbrace{\sum_{S_{t-1}} \underbrace{p(S_t|S_{t-1})}_{\text{Transition}} p(S_{t-1}|O_{t-1}))}_{\text{Belief}}, \tag{5.7}$$

where η is a normalization constant. The recursive estimation is being composed by two parts: firstly, the current state distribution is projected forward (prediction) from $t-1$ to t; then, it is updated using the new evidence O_t.

Prediction. Between $t-1$ and t, the posterior is updated according to the robot's motion through the transition model $p(S_t|S_{t-1})$, which is used to predict the distribution of S_t given each state of S_{t-1}. The combination of the above with the recursive part of the filter $p(S_{t-1}|O_{t-1})$ comprises the belief of the next event. Depending on the respective values of S_t and S_{t-1}, this probability is set with one of the following values, which are further discussed in Section 5.2.1:

- $p(S_t = No\ Loop \mid S_{t-1} = No\ Loop)$ = 0.975, the probability that no loop-closure event occurs at time t is high, given that no loop-closure occurred at time $t-1$.
- $p(S_t = Loop \mid S_{t-1} = No\ Loop)$ = 0.025, the probability of a loop-closure event at time t is low, given that no loop-closure occurred at $t-1$.
- $p(S_t = No\ Loop \mid S_{t-1} = Loop)$ = 0.025, the probability of the event "$No\ Loop$" at time t is low, given that a loop-closure occurred at time $t-1$.
- $p(S_t = Loop \mid S_{t-1} = Loop)$ = 0.975, the probability that a loop-closure event occurs at time t is high, given that a loop also occurred at time $t-1$.

Bayes Update. The sensor model $p(O_t|S_t)$ is evaluated using the locations' binomial probability score. Aiming to categorize the generated binomial scores into filter observations, the value range is split into two parts based on the probability threshold th:

$$p(O_t|S_t = No\ Loop) = \begin{cases} 1.00, \text{if } O_t > th, \forall l \in L. \\ 0.00, \text{if } O_t < th, \exists l \in L. \end{cases} \tag{5.8}$$

$$p(O_t|S_t = Loop) = \begin{cases} 0.46, \text{if } O_t > th, \forall l \in L. \\ 0.54, \text{if } O_t < th, \exists l \in L. \end{cases} \tag{5.9}$$

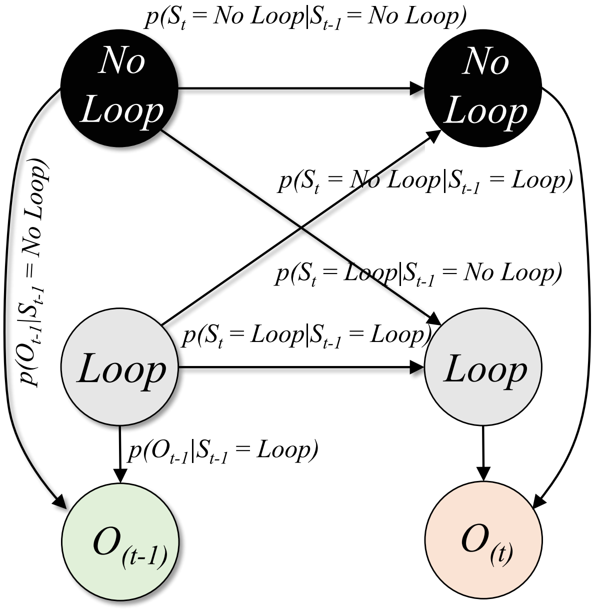

p(S_t = No Loop|S_{t-1} = No Loop)
No Loop
No Loop
p(S_t = No Loop|S_{t-1} = Loop)
p(S_t = Loop|S_{t-1} = No Loop)
p(O_{t-1}|S_{t-1} = No Loop)
p(S_t = Loop|S_{t-1} = Loop)
Loop
Loop
p(O_{t-1}|S_{t-1} = Loop)
O_(t-1)
O_(t)

Figure 5.3. *State machine representation of the proposed Hidden Markov Model (HMM) for loop-closure detection. Observations (O_{t-1}, O_t) are based on the system's binomial response $Pr(X_l(t) = x_l(t))$ among the database locations after the voting process. The light green observation indicates the existence of locations l which satisfy the binomial function's conditions ($\exists\, l \in L\ :\ O_t < th$), while the light orange examples correspond to the ones that do not ($\forall\, l \in L\ :\ O_t > th$).*

As shown, our observation model seeks into the set of locations L for the existence of database entries l which satisfy the binomial conditions. Notably, the system's initialization probabilities are set to a no loop-closure belief $p(S_0) = \langle 1, 0\rangle$, which derives from our confidence that such detection cannot occur at the beginning of any trajectory. The proposed model is summarized in the diagram in Fig. 5.3, while a discussion regarding the selected probability values is offered in Section 5.2.1.

5.1.2.4 A new or an old location?

Posterior, in the probabilistic context, means “after taking into account the relevant observation related to the examined cases". After $p(S_t|O_t)$ has been updated and normalized, the highest hypothesis is accepted as full posterior, that is, if the loop-closure hypothesis $p(S_t = No\ Loop \mid O_t)$ is higher than 50%, the system adds a new location to the database, otherwise a “*Loop*" is detected.

5.1.2.5 Location matching

Since the votes' distribution affects a group of consecutive images, the 10 most similar candidate loop-closing locations are considered among the database entries that satisfy the conditions in Section 5.1.2.2. In addition, when the perceived query camera measurement performs a loop in the trajectory, while none of the database observation scores satisfy the aforementioned conditions ($O_t > th, \forall l \in L$), a temporal consistency constraint is adopted so as to determine the candidates images. In order to cope with possible false positive detections, owed to potential perceptual aliasing in the environment (e.g., when different places contain similar visual cues), the selected camera frames are subjected to a geometrical check. In such a way, image pairs that cannot be correlated by a transformation matrix are rejected independently from their visual similarity. An image-to-image correlation is performed between the query I_Q and the accepted candidates. Computations are executed until a valid matrix is estimated through an ascending binomial score order.

Temporal Consistency. Let us consider that at time $t-1$, the system correctly indicates a previously visited location by matching pair $\langle I_{Q-1}, I_{M-1} \rangle$ and that at time t, the filter also indicates a loop; however, none of the locations satisfy the binomial threshold ($O_t > th, \forall l \in L$). The temporal constrain defines a group of images, which are the only set of database entries to be further examined as loop-closures. We determine this window as of size $2\kappa + 1$ centered around $I_M - 1$, i.e., $[I_{(M-1)-\kappa}, \ldots, I_{(M-1)+\kappa}]$. Nevertheless, locations which are not assigned with a binomial score are excluded.

Geometrical Verification. A fundamental matrix is estimated, through a RANSAC-based scheme, which is required to be supported by at least ϕ point inliers between the query I_Q and the matched image I_M. To compute these correspondences, tracked features are compared with the descriptors from the chosen location. A set of SURF descriptors are extracted, the cardinality of which is twice as big as the ones of each frame's TP, thus, offering an efficient balance between accuracy and computational complexity.

5.1.2.6 Visual vocabulary management

The goal of this process is to effectively handle the increasing rate of the vocabulary, which, until this stage, adds new elements no matter if a similar entry already exists. The objective is to remove multiple codewords of repetitive pattern representing the same environmental element at different time-stamps. On top of the database size and computational complexity reduction, this unmonitored development also results in a voting ambiguity. This issue is mostly evident when the agent revisits a certain route

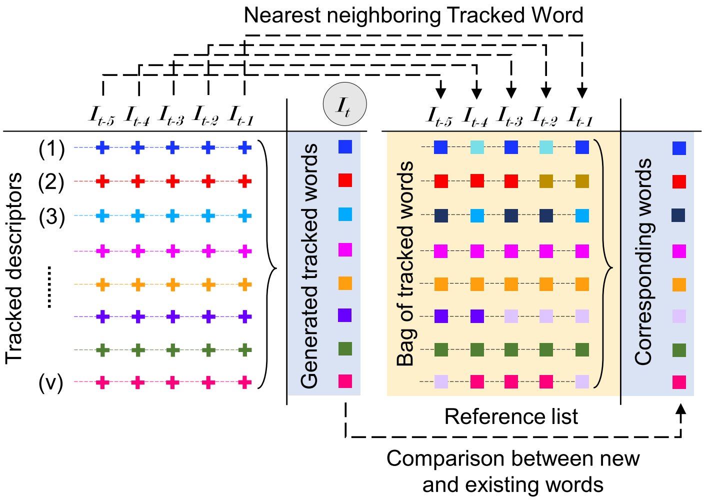

Nearest neighboring Tracked Word
I_{t-5} I_{t-4} I_{t-3} I_{t-2} I_{t-1}
I_t
I_{t-5} I_{t-4} I_{t-3} I_{t-2} I_{t-1}
Tracked descriptors
(1)
(2)
(3)
(v)
Generated tracked words
Bag of tracked words
Corresponding words
Reference list
Comparison between new and existing words

Figure 5.4. *The process of vocabulary management. As the trajectory escalates (..., $I_{t-3}, I_{t-2}, I_{t-1}, I_t$) along with voting procedure, a reference list regarding the tracked descriptors (block of crosses) and their nearest-neighboring tracked words (block of squares) is maintained. When the query location I_t is identified as a loop-closure, the most recently generated tracked words are checked with the most reported ones indicated via the reference list, in order to decide if they should be accumulated into the existing vocabulary.*

Algorithm 3 Vocabulary management.

```
Input: $I_Q$: Incoming image, $I_M$: Matched image, $Idx$: Location indexing list, $W_n$:
       Newly generated tracked word, $d_n$: Newly generated tracked word's descriptors,
       $RL$: Reference list
Output: $Idx$: Updated location indexing list, $BoTW$: Visual vocabulary
for each newly generated tracked word $W_n$ do
    $id$ = find(max($RL$, $W_n$)) // select the most voted word in database based on the
    $W_n$ descriptors' polling history
    $dist$ = norm($W_n$ - $W_{id}$) // euclidean distance
    $member$ = $Idx(id, I_M)$ // matched image contains most voted word
    if $dist$ < 0.4 and $member$ == true then
        $W_{id}$ = median($W_{id}$, $d_n$) // refresh the word's description
        $Idx$ = update($Idx$) // refresh the location indexing list based on the generated
        word's map position
    else
        $BoTW$ = add($W_n$) // add newly generated word since it doesn't exists in
        dictionary
    end
end
```

more than twice, in which case the query image will distribute votes from the same physical location to multiple ones, decreasing the system's discriminability.

Thus, during navigation, we create a reference list based on the matching process, which indicates tracked words being voted by the query descriptors. When a loop-closure is detected, each newly generated word is checked for a descriptors-to-word correspondence, to determine if the new element needs to be further processed or not. For each sequence of tracked descriptors, the most voted word in the database is indicated. Then, a similarity comparison based on equation 5.2 is applied on the chosen words' pair $\langle$newly generated, corresponding most voted$\rangle$, in which tracked words are considered to be similar if their distance is lower than 0.4. However, despite this check, the corresponding vocabulary's entry needs to satisfy a location condition check, meaning that the selected word is ignored if it is not associated with the chosen loop-closing image. Subsequently, tracked descriptors of the generated word and the one existing in the database are merged according to equation 5.3. Finally, the vocabulary's indexing list *Idx*, regarding the tracked word's locations, is updated to include the images corresponding to the merged word. A representative example is depicted in Fig. 5.4, while Algorithm 3 details this process.

5.2 Experimental results

Table 5.1. *Parameters utilized from the proposed single-based mapping pipeline.*

Parameter	Symbol	Value
SURF point response	Φ	400.0
Maximum # of Tracked Points	ν	150
Minimum points' distance	α	5
Minimum descriptors' distance	β	0.6
Minimum tracked word's length	ρ	5
Minimun RANSAC inliers	ϕ	8
Temporal consistency	κ	8

This section presents the experimental methodology followed to evaluate the proposed framework through an extensive set of tests on several datasets using the precision-recall metrics. Since the proposed pipeline needs to be adaptable to a variety of different environments, the nine image-sequences discussed in chapter 2 are selected for our experimental protocol. Among them, aiming to adjust the parameters of the algorithm, three are used for our evaluation. More specifically, the KITTI courses 00 and 05 are the ones selected for evaluating the vocabulary's evolution size as they provide long trajectories wherein loops are frequently presented. The third evaluation dataset is the Lip 6 outdoor. An important characteristic of this set is the fact that the camera visits some of the recorded locations more than twice, making it ideal for the assessment of the proposed vocabulary management mechanism. The performance of the presented approach is compared against the aforementioned methods, as well as the existing state-of-the-art techniques which are based on an incrementally generated visual vocabulary and a pre-trained one. BoTW-LCD is configured by using the parameters summarized in Table 5.1, which are extensively evaluated in Section 5.2.2.

5.2.1 Parameters' discussion

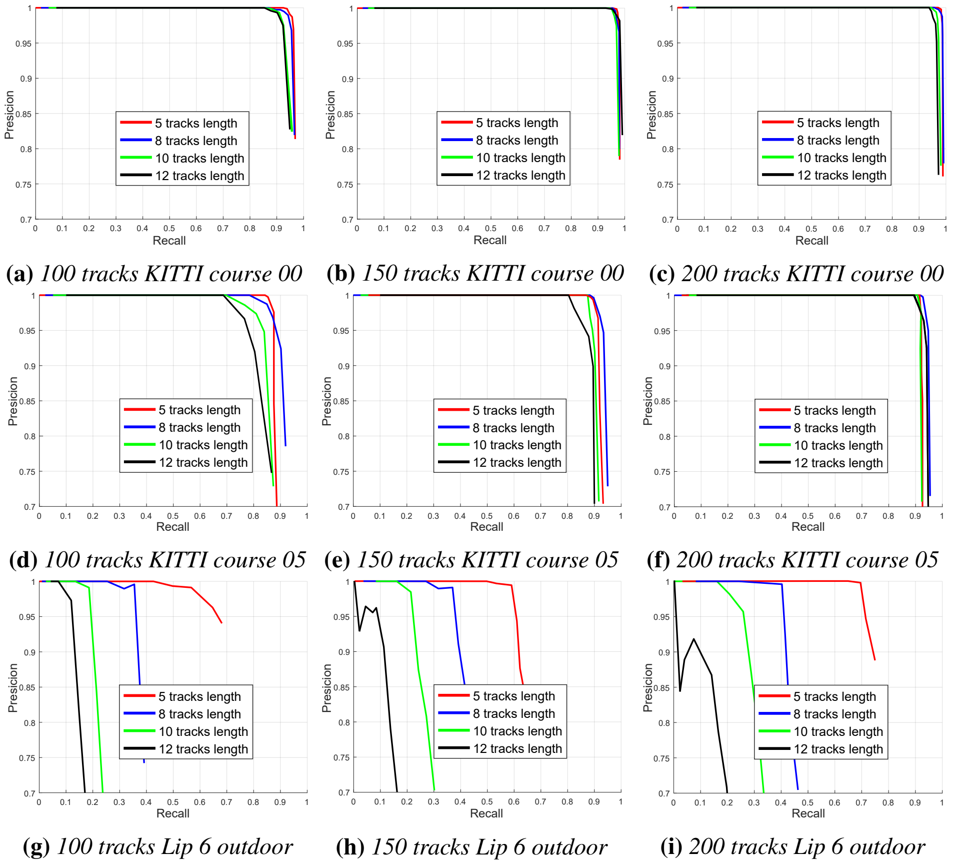
In this subsection, we briefly discuss the temporal parameters. In general, the performance of BoTW-LCD relies on the transition $p(S_t|S_{t-1})$ and observation $p(O_t|S_t)$ probabilities (see Section 5.1.2.3). The framework in which we work is quite simple, including two states for the transition model and the observation. The transition probabilities follow the loop-closure principle which indicates that a belief state would follow its previous one. Thus, their values are appropriately attributed to almost 98$\%$ in both cases (see Section 5.1.2.3). The observation model is the one which plays the primary role in shaping the final decision. Aiming to highlight the probabilities produced by the binomial density function, we have chosen a high level of confidence $p(O_t|S_t = No\ Loop) = 0.00$ when the loop-closure threshold is satisfied (equation 5.8) since its efficiency in identifying pre-visited locations has been well-established in Chapter 3. On the contrary, to avoid losing a possible detection in a sequence of loop events, due to the lack of satisfying the condition if $O_t > th, \forall l \in L$, its probability is defined at 46$\%$ (equation 5.9), allowing the system to correct its belief in the following observations while maintaining its high performance. These parameters are estimated empirically while a level of confidence about their values is attributed through the Hidden Markov Model (HMM) estimation algorithm proposed by [267]. All our experiments were performed under the same set of probability filtering values.

5.2.2 Performance evaluation

We illustrate the precision-recall rates for different cases of maximum retained tracked features (ν = 100, 150, 200). In addition, we assess the tracked words' minimum allowed length ($\rho = \{5, 8, 10, 12\}$), their merging approach (mean, median), the description method accuracy (SURF - 64D, SURF - 128D), as well as the vocabulary management for the achieved performance.

5.2.2.1 A modest vocabulary loop-closure detection

Aiming to evaluate the minimum required length for a tracked word generation, in Fig. 5.5 we present the precision-recall curves for a baseline version of the proposed system. In particular, we avoid the utilization of the proposed vocabulary management technique and the Bayes filter, while the 64D version of SURF is employed as *feature*



(a) *100 tracks KITTI course 00* **(b)** *150 tracks KITTI course 00* **(c)** *200 tracks KITTI course 00*

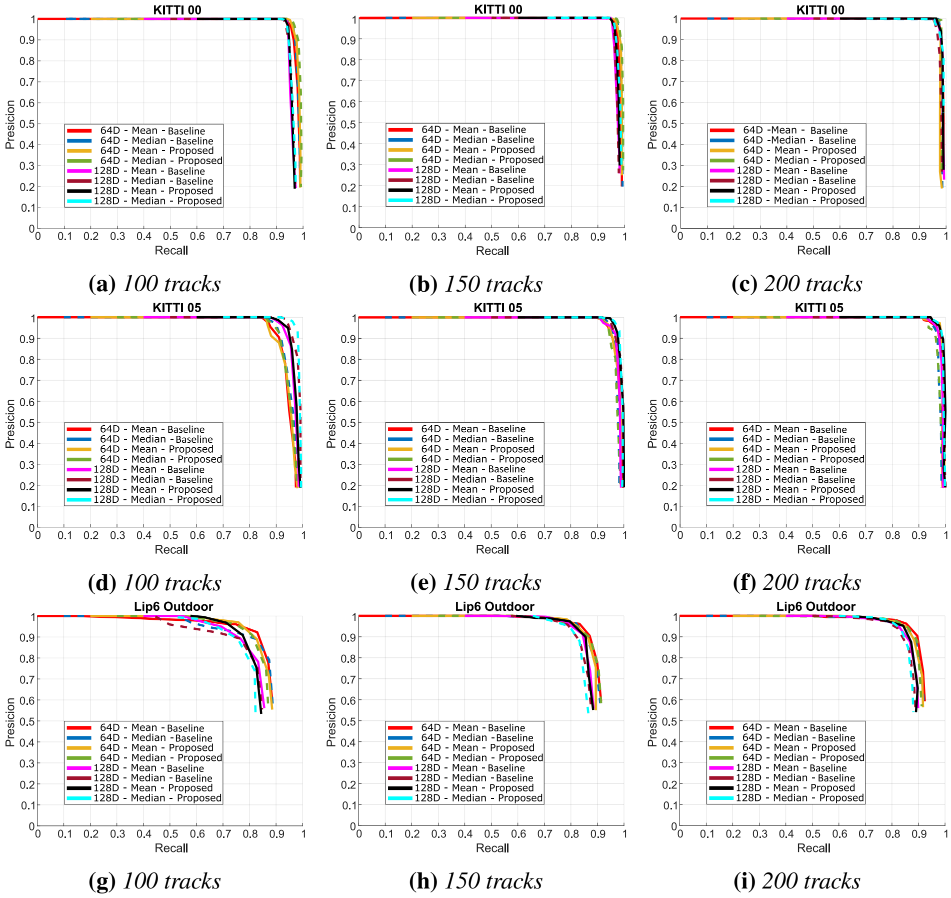
(d) *100 tracks KITTI course 05* **(e)** *150 tracks KITTI course 05* **(f)** *200 tracks KITTI course 05*

(g) *100 tracks Lip 6 outdoor* **(h)** *150 tracks Lip 6 outdoor* **(i)** *200 tracks Lip 6 outdoor*

Figure 5.5. *Precision-recall curves evaluating the minimum required length for a tracked word generation. Tests are performed on the KITTI courses [250], 00 (top), 05 (middle) and Lip 6 outdoor [140] (bottom) for the baseline version of the proposed method. As the number of Tracked Points (TP) grows, the performance is increased (recall rates for 100% precision) until it settles in the cases of 200 and 250. On the contrary, as the minimum allowed TW-length increases the system's performance constantly decreases.*

extraction module and the mean for the tracked words' generation. As observed, the recall rate (R_{P100}) increases with the number of tracked points, reaching more than 90% in KITTI courses and almost 70% in Lip 6 outdoor. However, in Lip 6 outdoor, where the acquisition rate is too low (1Hz), counter to the points' quantity the performance decreases as the tracked words' length gets longer, intensively reveling the effect of the lengthiness of tracked words. This is owed to the fact that points which appear for a short time-period in the trajectory are discarded from the BoTW reducing the potential of a richer vocabulary and a more accurate voting procedure.

Subsequently, we keep deactivated the temporal filter and the geometrical verification check in order to evaluate the proposed visual vocabulary management technique for each of the aforementioned cases in Fig. 5.6. Our first remark is that each of the



(a) *100 tracks* **(b)** *150 tracks* **(c)** *200 tracks*

(d) *100 tracks* **(e)** *150 tracks* **(f)** *200 tracks*

(g) *100 tracks* **(h)** *150 tracks* **(i)** *200 tracks*

Figure 5.6. *Precision-recall curves evaluating the number of maximum tracked features ν against the baseline approach and the proposed one using the vocabulary management technique. For each version, the tracked word generation method is presented along with the different descriptor version (64 & 128 dimension space of speeded-up robust features (SURF) [86]). Experiments are performed on the KITTI courses [250], 00 (top), 05 (middle) and Lip 6 Outdoor [140] (bottom). The proposed system seems to offer higher discrimination at voting procedure permitting similar recall rates for 100% precision between the cases of 150 and 200. The 128D SURF descriptors exhibit their robustness when a lower amount of tracked features is used as depicted in every of the evaluated dataset for the case of 100 tracks.*

Table 5.2. *Total of generated tracked words for each evaluated dataset.*

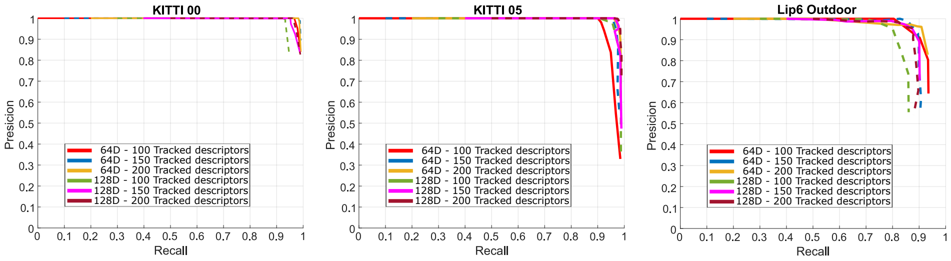
		Baseline (64D)	Proposed (64D)	Baseline (128D)	Proposed (128D)
Dataset	Tracked Points	Description method Mean / Median	Description method Mean / Median	Description method Mean / Median	Description method Mean / median
KITTI course 00	100	25603 / 25603	22930 / 22898	23577 / 23577	21092 / 22160
KITTI course 00	150	38170 / 38170	34196 / 34170	34951 / 34951	31514 / 31722
KITTI course 00	200	50510 / 50510	45741 / 45731	45976 / 45976	41669 / 41989
KITTI course 05	100	14853 / 14853	13432 / 13431	13832 / 13832	12448 / 12508
KITTI course 05	150	22199 / 22199	20135 / 20100	20515 / 20515	18548 / 18642
KITTI course 05	200	29391 / 29391	26728 / 26659	27009 / 27009	24516 / 24687
Lip 6 outdoor	100	4206 / 4206	3717 / 3724	3145 / 3145	2748 / 2788
Lip 6 outdoor	150	5776 / 5776	5066 / 5085	4309 / 4309	3768 / 3803
Lip 6 outdoor	200	6956 / 6956	6138 / 6236	5145 / 5145	4499 / 4576

produced curves presents high recall rates on the evaluation datasets. As one can observe, the system offers a very competent performance for 150 and 200 tracked features, approaching 95% recall in both KITTI courses, while keeping perfect precision. We observe that the median achieves similar performance to the mean-based. Furthermore, the 128D version of SURF shows higher recall rates for a lower number of tracked features, exhibiting its description accuracy for both the mean and median merging methods. In Lip 6 outdoor, which is evidently the most challenging image-sequence due to its low acquisition frame rate, visual resolution and rapid viewpoint variations, the recall extends to almost 90%, whilst maintaining high precision scores. It is notable that the recall curves corresponding to the proposed method, which incorporates vocabulary management, performs better in this dataset. This is owed to the fact that the specific image stream records the same route more than twice and the voting ambiguity originated from the arbitrary generation of new words is avoided.

In support thereof, we present a quantitative evaluation of the generated words in Table 5.2. Since our management technique is affected by the system's performance to detect loop-closures, the recorded number of words is obtained for the highest recall rate at 100% precision. A words' reduction of about 10% is observed for each case (ν = 100, 150, 200) for both merging methods and descriptors dimensions. In addition, more words are ignored as the number of tracked features increases, indicating that a higher number of elements are generated and remain in the database affecting the system's discrimination capabilities. Regarding the mean and the median versions, the results show a similar output with small fluctuations. Finally, although the description accuracy for the 128D version of SURF offers a lower amount of tracked words, we argue that this fact is not decisive for our approach since its memory footprint would be double the size of the 64D one.

5.2.2.2 Bayesian filtering

We now we present the evaluation of the Bayesian filtering approach which uses temporal context in Fig. 5.7. To exhibit the method's performance based on the binomial probability density function and Bayes filter, we illustrate the precision-recall rates following the same methodology for different loop-closure threshold th values. The experiments have been performed on the same evaluation datasets using the median approach for generating tracked words, while the geometrical verification was not activated. As we gradually evaluate individual frames, the posterior probability for non-loop and all possible loop-closure events at each query location is evaluated based on the loop hypothesis in Section 5.1.2.4. Table 5.1 presents the parameters selected in order to achieve a reduced computational complexity, while still preserving increased recall rates. Concerning the overall performance, we observe that an improved score is achieved by the BoTW-LCD in every image-sequence, reaching high recall rates at 100% precision. However, it is notable that in Lip 6 outdoor, an improvement in performance permits the



(a) *100 tracks* **(b)** *150 tracks* **(c)** *200 tracks*

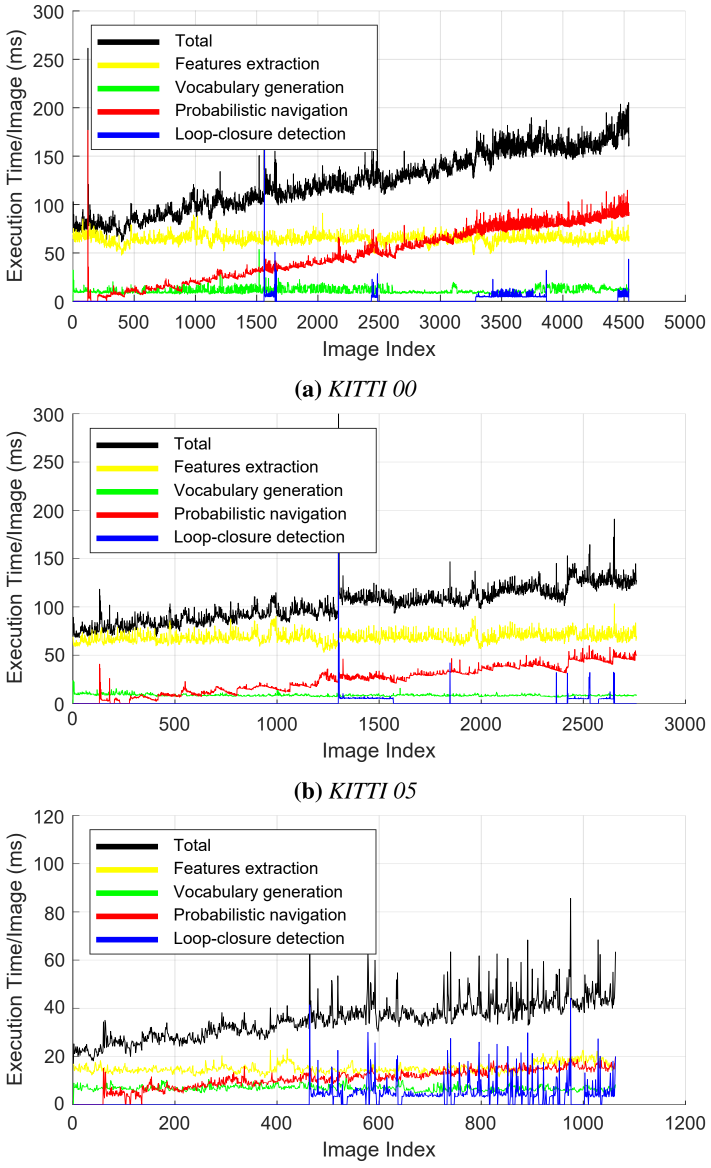
Figure 5.7. *Precision-recall curves evaluating the proposed system's performance through the utilization of the Bayes filter. Using the median approach during the generation of tracked words, precision and recall curves are illustrated for different speeded-up robust features' dimensions (64 & 128) [86] and maximum number of tracked features ν. High recall rates are obtained for each evaluated image stream. This is owed to the exploitation of the visited locations' temporal consistency along the navigation route in combination with the binomial probabilistic scoring.*

system to reach a score of about 85% when 150 tracks are employed. When the binomial score does not satisfy condition 5.5, temporal consistency prevents the system from detecting false-positive events in a different, though similar, area than the one where the previous loop-closure occurs. This way, a higher recall score is attained for both descriptor versions, allowing us to avoid the 128D method since it is computation-wise and memory-wise demanding.

5.2.3 System's response

The method's average timing results per image are shown in Fig. 5.8. To measure the execution time, we ran our framework on each of the evaluation datasets. Among them, the KITTI course 00 set is the longest one exhibiting a remarkable amount of loop-closure events. For this group of experiments, a total of 4551 images is processed, yielding 126.2 ms per query image on average. Table 5.3 shows an extensive timing documentation for each stage. The *feature extraction* process involves the computation of SURF key-points detection and description, while the *environment representation*, which corresponds to the visual vocabulary generation, is split into three steps: the key-points' tracking through the KLT method, the guided feature selection, and the tracked words' merging. The *decision making* is split into two steps: the probabilistic navigation which includes the exhaustive database search and the binomial probabilistic score computations, and the loop-closure detection step including the time required for the verification step through the calculation of the corresponding fundamental matrices, and the words' update due to the vocabulary management.

The results in Table 5.3 show that we can reliably detect loops in datasets that expand for 11 km while maintaining low execution times. We observe that all the involved steps are notably fast, considering the fact that we utilize a floating-point descriptor through



(a) *KITTI 00*

(b) *KITTI 05*

(c) *Lip 6 Outdoor*

Figure 5.8. *Execution time per image of the KITTI courses [250] 00, 05, and Lip 6 outdoor [140] for each of the main processing stages of the proposed algorithm.*

the SURF algorithm. BoTW-LCD is able to rapidly process images using a reduced set of visual words due to its innovative visual word management process. In contrast to the *binomial scoring*, the *database search* stage exhibits the highest execution time, due to the lack of an indexing scheme, followed by the *feature* extraction stage, which is known as the bottleneck point for many loop-closure approaches. The execution time for

Table 5.3. *Processing time per image (ms/query) of BoTW-LCD.*

		Average Time (ms)		
		KITTI course		Lip 6
		00	05	outdoor
Feature extraction	Key-point detection	41.4	45.5	7.6
	Key-point description	21.0	23.0	7.4
Environment representation	Key-point tracking	8.9	6.4	5.6
	Guided feature selection	2.0	2.0	1.1
	Merging words	2.9	2.6	1.5
Decision making	Database search	46.4	23.4	9.6
	Binomial scoring	0.8	0.8	1.0
	Geometrical verification	1.3	1.0	2.7
	Vocabulary management	1.5	0.6	2.2
Whole pipeline		126.2	105.3	38.7

environment representation is highly depended on the number of points and the tracker's parameters (e.g., pyramid levels, neighborhood area, maximum bidirectional error), while the required time for the *guided feature selection* and the *words' merging* is low. The *geometrical verification* stage and the *vocabulary management* are also negligible. As shown in Fig. 5.8, the proposed system achieves to estimate a valid fundamental matrix very fast reaching a value of 3 computations, on average, between the query I_Q and the accepted candidates.

5.2.4 Comparative results

This section extensively compares BoW-LCD against its baseline approach, as well as other modern solutions. In this regard, Table 5.4 contains the final mapping size SURF (#), the maximum recall at 100% of precision R(%) and the average response time per image T(ms) obtained for the proposed approach and its baseline version for every dataset. The performance of our system is measured by using a generic loop-closure threshold of $th = 2^{-9}$, which was obtained by the precision-recall curves in Fig. 5.7. This value is selected since it allows the system to achieve high recall rates in every evaluation dataset. During this experiment, our geometrical verification and vocabulary management modules are active, while the parameters remain constant so as to evaluate the adaptability of the approach. As can be observed, the impact in terms of recall is minimum and, in general, quite similar. However, BoTW-LCD is able to process an image in lessen time using a reduced set of visual words. We argue that this fact is mainly

Table 5.4. *In depth comparison with the baseline version of the proposed method.*

	Baseline version			BoTW-LCD		
Dataset	SURF(#)	R(%)	T(ms)	SURF(#)	R(%)	T(ms)
KITTI course 00	51K	97.5	173.5	**34K**	**97.7**	**126.2**
KITTI course 02	52K	80.0	190.2	**37K**	**81.5**	**133.0**
KITTI course 05	29K	92.6	130.1	**20K**	**94.0**	**105.3**
KITTI course 06	12K	98.1	98.7	**8K**	**98.1**	**90.1**
Lip 6 outdoor	7K	50.0	37.1	**5K**	**78.0**	**38.7**
EuRoC MH 05	20K	83.7	90.8	**13K**	**85.0**	**82.6**
Malaga parking 6L	41K	85.0	171.8	**28K**	**85.2**	**146.7**
New College	18K	83.0	82.1	**10K**	**87.0**	**67.5**
City Centre	3K	20.0	65.0	**2K**	**36.0**	**68.4**

Table 5.5. *In depth comparison with the work of Gehrig et al.*

	Gehrig *et al.*			BoTW-LCD		
Dataset	SURF(#)	R(%)	T(ms)	SURF(#)	R(%)	T(ms)
KITTI course 00	681K	92.8	920.3	**34K**	**97.7**	**126.2**
KITTI course 02	699K	80.2	990.7	**37K**	**81.5**	**133.0**
KITTI course 05	414K	86.0	572.8	**20K**	**94.0**	**105.3**
KITTI course 06	165K	**98.5**	185.9	**8K**	98.1	**90.1**
Lip 6 Outdoor	159K	**85.5**	232.9	**5K**	78.0	**38.7**
EuRoC MH 05	340K	53.8	310.4	**13K**	**85.0**	**82.6**
Malaga parking 6L	520K	64.0	770.0	**28K**	**85.2**	**146.7**
New College	394K	84.7	672.6	**10K**	**87.0**	**67.5**
City Centre	183K	**74.0**	232.7	**2K**	36.0	**68.4**

due to the new visual word managing process. Note that a comparison with off-line BoW schemes regarding their respective complexities is not presented since a direct analogy with methods based on a pre-trained vocabulary would not be meaningful. Following the results presented in Table 5.4, Fig. 5.9 illustrates the detections provided by BoTW-LCD at 100% precision for each image-sequence. The top path of each dataset presents the corresponding ground truth, that is, the trajectory which should be recognized in case the framework detects every loop-closure. When a loop is detected, the image triggering this event is labeled by a blue cycle. Note that in most cases, the loops are successfully detected, especially in the courses of the KITTI dataset.

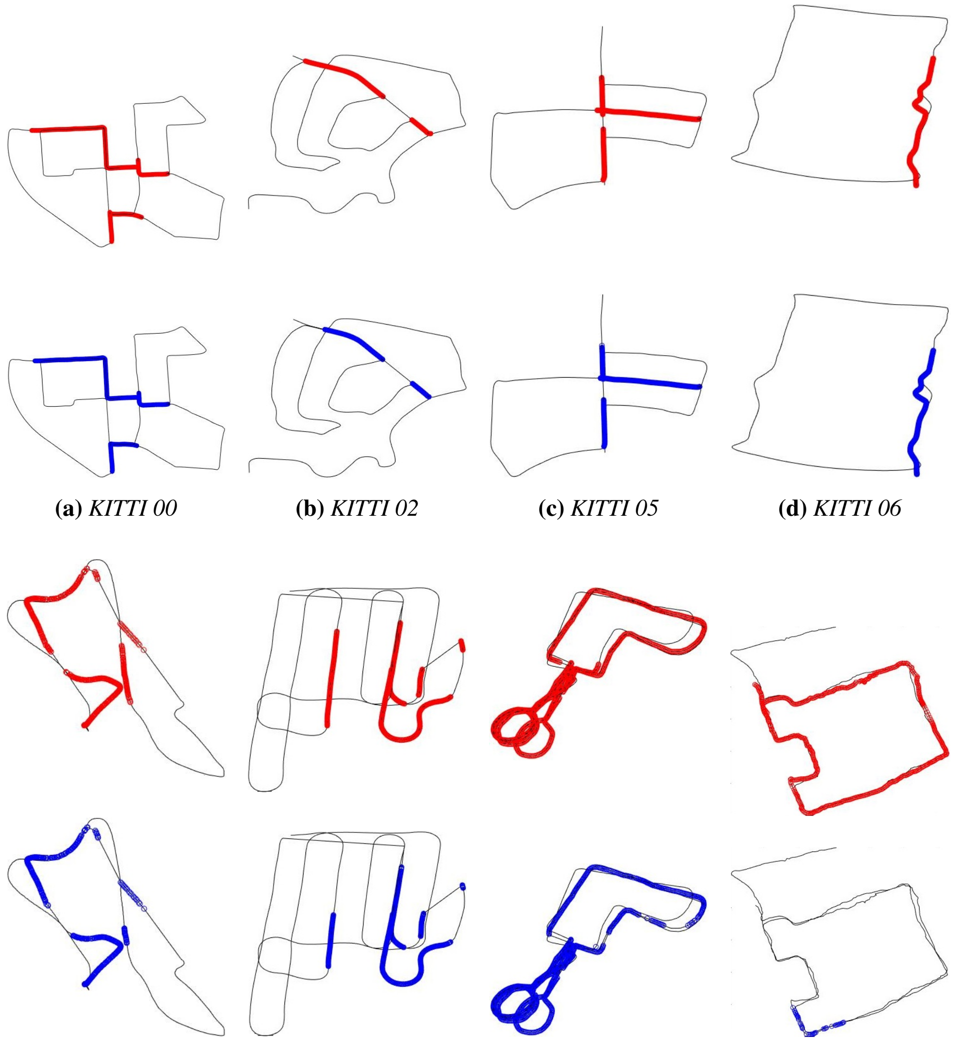
Subsequently, aiming to offer a more thorough view about the impact of our mapping technique, we compare the proposed pipeline against the SURF-based work of Gehrig *et al.* in Table 5.5. This version utilizes the same amount of SURF elements as the Tracked

Table 5.6. *In depth comparison with the framework of iBoW-LCD.*

	iBoW-LCD			BoTW-LCD		
Dataset	ORB(#)	R(%)	T(ms)	SURF(#)	R(%)	T(ms)
KITTI course 00	958K	76.5	400.2	**34K**	**97.7**	**126.2**
KITTI course 02	950K	72.2	422.3	**37K**	**81.5**	**133.0**
KITTI course 05	556K	53.0	366.5	**20K**	**94.0**	**105.3**
KITTI course 06	212K	95.5	385.1	**8K**	**98.1**	**90.1**
Lip 6 outdoor	121K	**85.2**	228.0	**5K**	78.0	**38.7**
EuRoC MH 05	443K	25.6	350.4	**13K**	**85.0**	**82.6**
Malaga parking 6L	806K	57.4	440.8	**28K**	**85.2**	**146.7**
New College	254K	73.1	383.7	**10K**	**87.0**	**67.5**
City Centre	67K	**88.2**	336.2	**2K**	36.0	**68.4**

Points ν used by the proposed method to describe the incoming frame. Furthermore, a 40 sec temporal window is included for rejecting early visited locations similar to the one used in Chapter 3. Accordingly, for searching the database and aggregating votes, $k = 1$ nearest-neighbor is selected, while the parametrization of the geometrical check between the chosen pair is also based on the proposed work. The best-performing loop-closure threshold for each assessed case is evaluated according to the literature and the selected parameters remained constant over all datasets. Next, in Table 5.6 we compare the proposed pipeline with the iBoW-LCD framework, which is based on binary codewords for generating the vocabulary. Notice the high reduction of the final mapping size (SURF against ORB) and the timings offered by the proposed approach in comparison to the other methods. Nevertheless, as shown in both Tables 5.5 and 5.6, building a map through tracked words does not always imply higher recall values. However, it consistently reduces the computational times and the size of the final map. It is worth to mention that both in Lip 6 outdoor and City Centre, which are two challenging image-sequences (e.g., due to the cameras' orientation), the other approaches perform better since are tend to work as image-retrieval methods having a distinct representation for each incoming frame. Moreover, with the aim to enrich the comparative analysis, Table 5.7a presents the memory consumption in each trajectory mapping for some of the most acknowledged methods that aim for a real-time and lightweight implementation. As shown, BoTW-LCD achieves the lowest footprint in every dataset. Note that the low memory usage of the iBoW-LCD vocabulary is mainly due to its binary form. Similarly, PREVIeW, which uses a binary dictionary of 1M visual words, utilizes only 30.5 Mb of memory.

Furthermore, in Table 5.7b our approach is compared against the most representative works in visual place recognition which are independent from any training stage, namely IBuILD, Kazmi and Mertsching, FILD, as well as our previously presented approaches.



(a) *KITTI 00* **(b)** *KITTI 02* **(c)** *KITTI 05* **(d)** *KITTI 06*

(e) *EuRoC MH 05* **(f)** *Malaga 6L* **(g)** *New college* **(h)** *City Centre*

Figure 5.9. *Loop-closures generated from the proposed pipeline for every evaluated dataset using the parameters defined in Table 5.1. In each trajectory, red cycles indicate ground truth information, while the blue ones illustrate the system's detections. The top row presents the KITTI courses [250] 00, 02, 05 and 06, whilst EuRoC MH 05 [251], Malaga 6L [252], New College [254], and City Centre [129] are depicted in the bottom row. As can be seen in most of the cases, BoTW-LCD achieves to recognize locations when the robot traverses a route which presents similar visual content. This is especially highlighted in the KITTI datasets, where the frames are captured from a forward facing camera, in contract to City Centre's lateral sensor orientation.*

(a) *Memory usage comparison with different state-of-the-art systems. Bold values indicate minimum consumption per evaluated dataset.*

	KITTI course				Lip 6	EuRoC	Malaga	New College	City Centre
	00	02	05	06	outdoor	MH 05	parking 6L		
Method	S(Mb)	S(Mb)	S(Mb)	S(Mb)	S(Mb)	S(Mb)	S(Mb)	S(Mb)	S(Mb)
Baseline version	12.4	12.6	7.0	2.9	1.7	4.8	10.0	4.3	0.7
iGNGmap-LCD	11.0	11.2	6.1	2.5	2.0	4.5	7.5	8.0	0.6
Gehrig *et al.*	166.2	170.6	101.0	40.2	38.8	83.0	126.9	96.1	44.6
iBoW-LCD	29.2	28.9	16.9	6.4	3.6	13.5	24.5	7.7	2.8
PREVIeW	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5
BoTW-LCD	**8.3**	**9.0**	**4.8**	**1.9**	**1.2**	**3.1**	**6.8**	**2.4**	**0.5**

(b) *Performance comparison with other well-known appearance-based place recognition methods. Bold values indicate maximum performance per evaluated dataset.*

	KITTI course				Lip 6	EuRoC	Malaga	New College	Oxford
	00	02	05	06	outdoor	MH 05	parking 6L		City Centre
Off-line approaches									
FAB-MAP2.0	61.2	44.3	48.5	64.5	N/A	N/A	21.8	52.6	40.1
DBoW2	72.4	68.2	51.9	89.7	N/A	N/A	74.7	47.5	30.6
PREVIeW	96.5	72.0	**97.3**	80.1	58.3	23.1	33.9	80.8	71.1
Online approaches									
IBuILD	92.0	N/A	N/A	N/A	25.6	N/A	78.1	N/A	38.9
Kazmi and Mertsching	90.3	79.4	81.4	97.3	**84.9**	26.8	50.9	51.0	75.5
FILD	91.2	65.1	85.1	93.3	0.3	–	56.0	76.7	66.4
iGNGmap-LCD	93.1	76.0	94.2	86.0	12.0	69.2	87.9	88.0	16.3
Tracking-DOSeqSLAM	77.6	61.1	38.2	–	40.9	–	42.0	40.0	47.1
BoTW-LCD	**97.7**	**81.5**	94.3	**98.1**	78.0	**85.0**	**87.9**	**89.2**	36.0

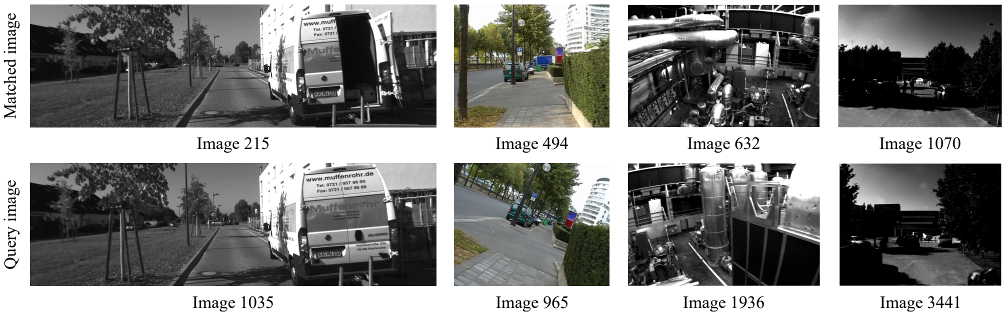


Figure 5.10. *Example images which are correctly identified by the proposed framework as loop-closure detections. The query frame I_Q is the image recorded by the robot at time t, whereas the matched frame I_M corresponds to the chosen location. From left to right: KITTI course 02 [250], Lip 6 outdoor [140], EuRoC MH 05 [251] and Malaga 6L [252].*

In addition, for the sake of completeness, comparisons are also given against pipelines based on a previously trained vocabulary with the aim to help the reader to identify the place of the proposed pipeline within the state-of-the-art. More specifically, FAB-MAP 2.0, DBoW2, and PREVIeW are chosen. By examining Table 5.4, one can observe the significantly high score achieved by our method in the Lip 6 outdoor dataset. We succeed to excel among the baseline version, highlighting the importance of the temporal information across the trajectory. Nonetheless, the proposed framework performs unfavorably against iBoW-LCD and Kazmi and Mertsching as shown in Table 5.7b. This is due to the geometrical verification parameterization which strengthens our pipeline's accuracy in the cost of missing some of its potential performance, but also due to the fact that the system encounters a route of low textured images (avg. features/image), impairing our feature tracking procedure. This characteristic drops the recall rate when the geometrical verification step is applied since some of the true-positive detections are discarded as they fail to produce a valid fundamental matrix with enough inliers. However, we also need to stress out that our method's performance is able to reach even higher recall rate than the ones in Table 5.7b (as illustrated in the precision-recall curves in Fig. 5.7), yet our aim is to present a system with a homogeneous set of parameters that can be used in any environment. Thus, the adopted probability threshold and the RANSAC inliers are selected and fixed so as to maintain high scores for 100% precision across every evaluated dataset. In the KITTI course 00 set, the proposed framework exhibits over 97% of recall results, while compared to the rest of the sequences of the KITTI suite, it outperforms most of the competitors. Moreover, in the testing cases, our framework demonstrates a significant improvement to the obtained recall. In EuRoC MH 05, Malaga 6L, and New College a score of 85% is reached on each dataset, while holding a high precision rate. It is noteworthy that in EuRoC MH 05, where the system confronts an environment of low illumination, the binary description methods, adopted in PREVIeW and iBoW-LCD, are unable to perform competitively compared to the floating point features. Similarly, global descriptors utilized in FILD and Tracking-DOSeqSLAM present low recall scores. This results in a high divergence in terms of recall scores against the proposed pipeline. Finally, in the case of City Centre, our system fails to follow the performance of the other solutions. This fact implies that our mapping procedure performs better when the camera's orientation is frontal, allowing the formulation of prolonged word tracks. In Fig. 5.10, some accurately detected locations are shown.

6 Open challenges and conclusion

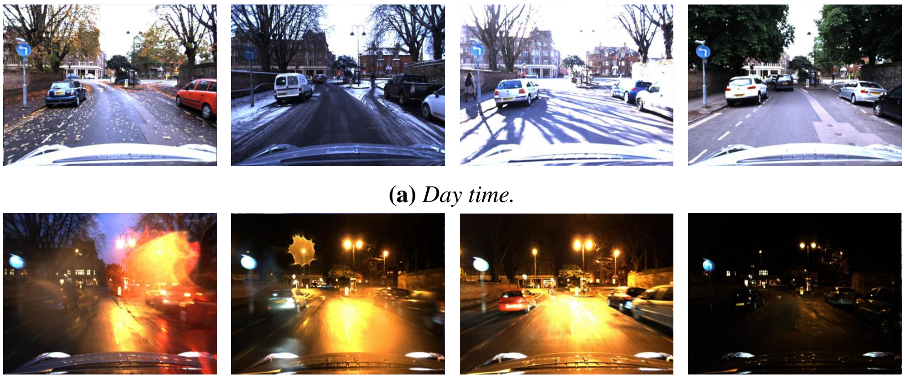
6.1 New challenges: Long-term operation

As presented in this thesis, the main objective of any loop-closure detection pipeline is to facilitate robust navigation for an extended period and under a broad range of viewing situations. Moreover, within long-term and large-scale SLAM autonomy, previously visited locations in dynamic environments need to be recognized under different day periods and scenes with changeable illumination and seasonal conditions [80, 98, 233]. As a result, it becomes increasingly difficult to match two images, mainly since such variations affect the image appearance significantly (Fig. 6.1). Furthermore, extreme viewpoint variations lead to severe perspective distortions and low overlap between the query and the database frames.

Another critical aspect in long-term applications is the storage requirements needed to map the whole environment effectively since the majority of approaches scale linearly to the map's size (at best). Consequently, there has been much interest in developing compact appearance representations so as to demonstrate sub-linear scaling in computational complexity and memory demands. These techniques typically trade off memory usage with detection performance, or vice versa, for achieving computational efficiency as shown by the proposed method in Chapter 4.

6.1.1 Dynamic environments

During navigation in a changing environment, the topological information about the robot's relative movement becomes more important as noise from the sensory inputs is accumulated to an overwhelming degree [269, 270]. Early works exploited the topological information through sequence matching [110, 232], or network flows [215]. However, their output is still dependent on their visual representations' quality since the utilized hand-crafted features are not distinctive enough so as to form a genuinely



(a) *Day time.*

(b) *Night time.*

Figure 6.1. *Example images from the Oxford RobotCar dataset [268] for both (a) day-time and (b) night-time conditions. From left to right: Autumn, winter, spring, and summer. Within long-term and large-scale SLAM autonomy, detections need to be successful despite significant variations in the images' context, such as different illumination conditions, e.g., day and night, or year seasons.*

reusable map [271]. On the contrary, representations provided by deep learning techniques show promising results on applications with challenging conditional and viewpoint changes [171, 189, 199, 272]. More specifically, deep learning approaches can be utilized to either construct description features with increased robustness to perceptual changes [273, 274] or to predict and negate the effect of appearance variations [156, 215, 275, 276]. It is also worth noting that for both the above cases, networks that are previously trained for semantic place classification [277] outperform the ones designed for object recognition when applied for place recognition under severe appearance changes [183].

6.1.1.1 Robust visual representations

Such techniques are mainly based on a single global descriptor. A series of works have been developed based on SeqSLAM following the same architecture [215, 278], that does not adopt learned features as their representation mechanism. Among the different variants, the gist-based pipeline [95] compares the learning-based ones [279, 280]. Another approach by Maddern and Vidas [281] utilize two different visual vocabularies by combining SURF-based visual words from the visible and infrared spectrum. Their results showed that hand-crafted features could not achieve high performances in complicated dynamic environments; however, the infrared data are more robust to extreme variations. On the other hand, techniques which are built upon learned features typically demand an extensive labeled training set [174, 176, 202, 282]; however, there exist some exceptions that do not require environment-specific learning samples [222].

6.1.1.2 Learning and predicting the appearance changes

These methods require labeled training data, such as matched frames from the exact locations under different conditions [275, 276]. An average description of images was learned, viz., a vector of weighted SIFT features in [283]. Their system was trained in summer and winter environments looking for valuable features capable of recognizing places under seasonal changes. The features that co-occurred in each image taken at different times of the day are combined into a unique representation with identifiable points from any point of view, irrespective of illumination conditions [221]. Similarly, matching observations with significant appearance changes is achieved using a support-vector machine (SVM) classifier to learn patch-based distinctive visual elements [274, 284]. This approach yields excellent performance but has the highly restrictive requirement that training must occur in the testing environment under all possible environmental conditions. The authors in [278] learned how the appearance of a location changes gradually, while Neubert *et al.* [275] constructed a map based on visual words originated from two different conditions. A super-pixel dictionary of hand-crafted features specific for each season is built in [285] by exploiting the seasonal appearance changes' repeatability. Using change-removal, which is similar to dimensionality reduction, showed that by excluding the less discriminative elements of a descriptor, an enhanced performance could be achieved [19,286]. Another way to tackle such challenges is based on illumination-invariant image conversions [156,287,288], and shadow removal [289, 290]. The former transfers images into an illumination invariant representation; however, it is shown that the hypothesis of a black-body illumination is violated, yielding poor results [156]. Shadow removal techniques were used to obtain invariant illumination images independent of the sun's positions.

Lategahn *et al.* [291] are the first to study how the CNNs can be used for learning illumination invariant descriptors automatically. Exploiting the visual features extracted from ConvNet [165], a graph-based visual loop detection system is proposed in [241], while a BoW for landmark selection is learned in [292]. Modifying images to emulate similar query and reference conditions is another way to avoid addressing the descriptors for condition invariance. The authors in [293] learn an invertible generator, which transforms the images to opposing conditions, e.g., summer to winter. Their network is trained to output synthetic images optimized for feature matching. Milford *et al.* [294] proposed a model to estimate the corresponding depth images that are potentially condition-invariant.

6.1.2 Viewpoint variations

Viewpoint changes are as critical as the appearance variations since visual data of the same location may seem much different when captured from other views [295]. The variation in viewpoint could be a minor lateral change or a much-complicated one, such as bi-directional or angular changes coupled with alterations in the zoom, base point, and focus throughout repeated traverses. Most pipelines are focused on unidirectional loop-closure detections. However, in some cases, they are not sufficient for identifying previously visited areas due to bidirectional loop-closures, i.e., when a robot traverses a location from the opposite direction. This type of problem is crucial because solely unidirectional detections do not provide robustness in long-term navigation. Pipelines, such as ABLE-P [296], identify bidirectional loops by incorporating panoramic imagery. A correspondence function to model the bidirectional transformation, estimated by a support-vector regression technique, is designed by the authors in [297] to reject mismatches. To achieve greater viewpoint robustness, semantically meaningful mapping techniques are adopted to detect and correct large loops [178, 202, 298]. Using visual semantics, extracted via RefineNet [299], multi-frame LoST-X [206] accomplished place recognition over opposing viewpoints. Similarly, appearance invariant descriptors (e.g., objects detected with CNN [165, 177, 188, 199, 205] or hand-crafted rules [171]) show that semantic information can provide a higher degree of invariability. Likewise, co-visibility graphs, generated from learned features, could boost the invariance to viewpoint changes [95, 168]. Finally, another research trend which has recently appeared tries to address the significant changes in viewpoint when images are captured from ground to aerial platforms using learning techniques. In general, the world is observed from much the same viewpoints over repeated visits in cases of ground robots; yet, other systems, such as a small UAV, experience considerably different viewpoints which demand recognition of similar images obtained from very wide baselines [20, 200].

6.1.3 Map management and storage requirements

As mentioned during this dissertation, scalability in terms of storage requirements is one of the main issues every autonomous system needs to address within the long-term mapping. In dense maps, in which every image is considered as a node in the topological graph, the loop-closure database increases linearly with the number of images. Consequently, for long-term operations that imply an extensive collection of images, this task becomes demanding not only to the computational requirements but also the system's performance. This problem is tackled through map management techniques: 1) using sparse topological maps, representing the environment with fewer nodes which correspond to visually distinct and strategically interesting locations (key-frames), 2) representing each node in a sparse map by a group of sequential and visually similar images, and 3) limiting the map's size by memory scale discretization.

6.1.3.1 Key-frame selection

This pipeline are based on the detection of scenes' visual changes by utilizing methods developed for video compression [300]. However, the main difference between key-

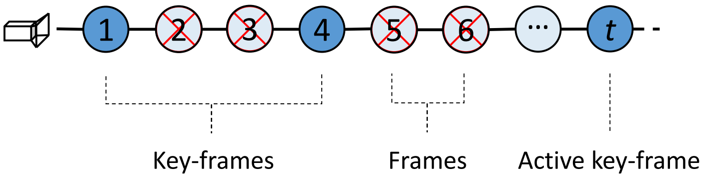


Figure 6.2. *Illustration of a map represented by key-frames.*

frame mapping and video abstraction is that the former requires the query image's localization with a previously visited location. This is vital for the system's performance since a single area might be recorded by two different locations [47]. Both locations may reach half of the probability mass, and therefore, neither attracts the threshold for successful data matching. Traditionally, the metric for deciding when to create graph nodes is typically an arbitrary one. Representative examples include the distance and angle between observations in space [179, 301], specific time intervals [302, 303], and a minimum number of tracked landmarks [228, 304–306]. An illustration of a map represented by key-frames is shown in 6.2.

6.1.3.2 Representing each node in a sparse map by a group of sequential and visually similar images

This category of techniques, wherein the presented method in Chapter 3 belongs, is a well-established process that offers computational efficiency while also retaining high spatial accuracy. Techniques that fall into this category map the environment hierarchically [307–311] and tackle scalability through the formulation of image groups, thus reducing the database's search space [149, 312–314]. Hierarchies have also been found in the mammalian brain, both in the structure of grid cells in the Hippocampus [315] and the visual cortex's pathway [316]. To limit the number of database instances, clustering [211, 220, 317, 318] or pruning [319] methods can be used and restrain map's parts which exceed a threshold based on the spatial density. Hierarchical approaches follow a two-stage process: firstly, less-intensive nodes are selected, and, next, the most similar view in the chosen node is searched [208, 320], as presented to our method. For instance, in [102], a hierarchical approach based on color histograms allows the identification of a matching image subset, and subsequently, SIFT features are utilized for acquiring a more precise loop-closing frame within this subset. Similarly, nodes are formulated by grouping images with common visual properties, represented by an average global descriptor and a set of binary features through on-line BoW [149].

6.1.3.3 Short-memory scale discretization

Limiting the map's size, so that loop-closure detection pipelines keep a processing complexity under a fixed time constrain and satisfy the online requirements in long-term operations. Mobile robots have limited computational resources; therefore, the map must be somewhat forgotten. Nevertheless, this needs ignoring of locations, a technique that leads to mismatches in future missions. On the contrary, maintaining in random access memory the entire robot's visual history is also sub-optimal and, in some cases, not possible. Dayoub and Duckett [321] map the environment by using reference views, i.e., many known points. Two specific memory time scales are included in every view: a short-term and a long-term. Frequently observed features belonging in the short-term memory advance to the long-term memory, while the ones not frequently observed are forgotten. Following a similar process, real-time appearance-based mapping (RTAB-MAP) [145] use short-term and long-term memory, while the authors in [219] assumed a system that includes working memory and an indexing scheme built upon the coreset streaming tree [322]. The method in [183] encodes regularly repeating visual patterns in the environment.

6.1.4 Computational Complexity

In contrast to computer vision benchmarks, wherein the recognition accuracy constitutes the most crucial metric regarding performance measurement, robotics depends on flexible algorithms that can perform robustly under certain real-time restrictions. As most appearance-based loop-closure detection solutions share the concepts of feature extraction, memorization, and matching, storage and computational costs, which increase drastically with the environment size, constitute such systems' weaknesses [57,110,212]. Given the map management strategies mentioned in Section 6.1.3 for large-scale operations, the main constraints to overcome are the visual information storage and the complexity of similarity computations. If one is to take the naive approach of using an exhaustive nearest neighbor search and directly comparing all the visual features of the current robot view with all of those observed so far, the complexity of the approach would become impractical as presented in this this thesis through the implementation of Gehrig et al.. This is due to the comparisons performed for images that do not exhibit the same context. This gets regressively less feasible as the run-time is analogous to the size of previously seen locations. Therefore, compact representations [210,323] and hashing methods [211] have been explored, apart from data structure-based retrieval techniques, e.g., trees [263,265,324] and graphs [149,325,326]

As the computational time of feature matching varies according to the visual feature's length, encoding the data into compact representations reduces the storage cost and simultaneously accelerates the similarity computations [327]. Using the most discriminant information in high-dimensional data, Liu and Zhang [94] perform loop-closure detection based on a PCA technique. They achieve to reduce the descriptor space from 960 dimensions to the 60 most discriminative ones while preserving high accuracy. Another line of frameworks adopt binary descriptors to improve computational efficiency [131] or encoded the high-dimensional vectors into compact codes, such as hashing [328]. Typical matching techniques include hashing, e.g., locality sensitive hashing (LSH) [329] or semantic hashing [330]. Although LSH does not need any previously processing or off-line procedures [211, 331, 332], its discrete feature representations suffer from data collisions when their size is large [333]. Nevertheless, with a view to avoid data collision and achieve unique mapping, visual information is embedded in continuous instead of discrete lower-dimensional spaces [334]. Avoiding dimensionality reduction or binary feature vectors, many pipelines were based on GPU-enabled techniques to close loops in real-time with high efficiency [224, 253].

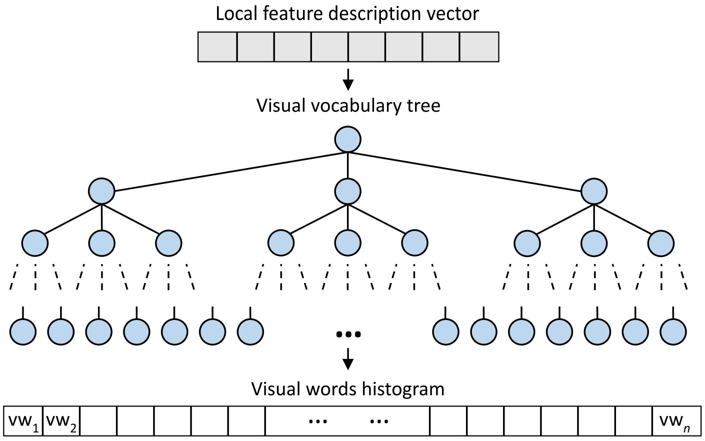


Figure 6.3. *The structure of a hierarchical visual vocabulary tree used in off-line visual bag of words pipelines [335]. Instead of searching the whole vocabulary to identify the most similar visual word, incoming local feature descriptors traverse the tree significantly reducing the required computations.*

Nister and Stewenius improve the indexing scheme of the off-line visual BoW model through a vocabulary tree generated via hierarchical k-means clustering [335], as depicted in Fig. 6.3. This way, faster indexing is achieved, while high performances are preserved [121, 210, 305]. Other works are based on spatial data structures [336] and agglomerative clustering [317]. The inverted multi-index [264] and different tree structures, e.g., k-d trees [262], randomized k-d forests [337], Chow Liu trees [338], decision trees [339]. More specifically, data structures, such as pyramid matching [340, 341], were used to detect loop-closures when high dimensional image descriptors were adopted [342, 343]. Furthermore, approaches based on the randomized k-d forest [70, 84, 106, 124, 263] are shown to perform better than a single k-d tree [66] or a Chow Liu tree [129]. It is worth noting that k-d trees are unsuitable when incremental visual vocabularies are selected since they become unbalanced if new descriptors are added after their construction [124]. Yet, this issue is avoided in off-line BoW models since their vocabulary is built a priori, and there is no other time-consuming module regardless of how large the map becomes.

Finally, although impressive outcomes have been achieved by utilizing deep learning, such approaches are yet computationally costly [202]. Increasing the network's size results in more computations and storage consumption at the time of training and testing. However, efforts to reduce their complexity do exist [159]. To bridge the research gap between learned features and their complexity, a CNN architecture employing a small number of layers previously trained on the scene-centric [344] database reduced the computational and memory costs [190]. Similarly, the authors in [272] compress the learned features' unnecessary data into a tractable number of bits for robust and efficient place recognition.

6.2 Conclusion

Closing this dissertation, the author wish to note that much effort has been put to produce efficient and robust methods to obtain accurate and consistent maps since the first loop-closure detection system. In this thesis, we addressed this problem with three modern solutions, which allowed the trajectory to be mapped through different methods, i.e., hierarchical, sequence- and single-based, while none of the aforementioned techniques requires a training step. To this end, concerning the ones adopting the BoW model, the visual vocabulary is fully adapted to each individual environment. As shown through our extensive experimentation on an entry-level system with an Intel Core i7-6700HQ (2.6 GHz) processor and 8 GB RAM, high recall scores are achieved for perfect precision in all assessed datasets outperforming existing state-of-the-art methods while maintaining low execution times and real-time behavior on routes as large as 13 km. Moreover, we showed that 64D SURF descriptors are able to outperform the 128D ones achieving higher accuracy and lower memory consumption, while using a vocabulary management technique when a previously visited location is detected improves the computational time and the system's accuracy. It is noteworthy that less than 37K visual words are totally produced for a route of 13Km using 8.3 Mb of memory, which is significantly shorter than any other cited work. Employing a probabilistic voting scheme when searching for previously visited locations into the map, a high degree of confidence about the images' similarity is achieved. Furthermore, a Bayes filter exploits the temporal aspect of the data gathered along the traversed path and finally, a geometrical verification step is performed to reject possible remaining outliers. In our future work, we intend to enhance the proposed pipelines with more sophisticated verification and indexing techniques to further increase the recall scores and reduced run-times.

Bibliography

[1] John O'Keefe and DH Conway, "Hippocampal place units in the freely moving rat: why they fire where they fire," *Experimental brain research*, vol. 31, no. 4, pp. 573–590, 1978. (page 1).

[2] Torkel Hafting, Marianne Fyhn, Sturla Molden, May-Britt Moser, and Edvard I Moser, "Microstructure of a spatial map in the entorhinal cortex," *Nature*, vol. 436, no. 7052, pp. 801–806, 2005. (page 1).

[3] Edvard I Moser, Emilio Kropff, and May-Britt Moser, "Place cells, grid cells, and the brain's spatial representation system," *Annual review of neuroscience*, vol. 31, pp. 69–89, 2008. (page 1).

[4] Richard Szeliski, *Computer vision: algorithms and applications*. Springer science and business media, 2010. (page 1).

[5] Shigang Li and Saburo Tsuji, "Selecting distinctive scene features for landmarks," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Nice, France, May 1992, pp. 53–54. (page 2).

[6] Benjamin Stewart, Jonathan Ko, Dieter Fox, and Kurt konolige, "The revisiting problem in mobile robot map building: a hierarchical Bayesian approach," in *proceedings of the conference on uncertainty in artificial intelligence (UAI)*, Acapulco, Mexico, August 2002, pp. 551–558. (page 2).

[7] Cheng Chen and Han Wang, "Appearance-based topological Bayesian inference for loop-closing detection in a cross-country environment," *The international journal of robotics research (SAGE)*, vol. 25, no. 10, pp. 953–983, 2006. (pages 2 and 17).

[8] Dean A Pomerleau, "ALVINN: an autonomous land vehicle in a neural network," in *proceedings of the international conference on neural information processing systems (NIPS)*, Denver, CO, USA, January 1988, pp. 305–313. (page 3).

[9] ——, *Neural network perception for mobile robot guidance*. Springer science & business media, 2012, vol. 239. (page 3).

[10] Yong Nyeon Kim, Dong Wook Ko, and Il Hong Suh, “Visual navigation using place recognition with visual line words,” in *proceedings of the international conference on ubiquitous robots and ambient intelligence (URAI)*, Kuala Lumpur, Malaysia, November 2014, pp. 676–676. (page 3).

[11] Stephan Weiss, Markus W Achtelik, Simon Lynen, Michael C Achtelik, Laurent Kneip, Margarita Chli, and Roland Siegwart, “Monocular vision for long-term micro aerial vehicle state estimation: A compendium,” *Journal of field robotics (Wiley)*, vol. 30, no. 5, pp. 803–831, 2013. (page 3).

[12] Shaowu Yang, Sebastian A Scherer, Xiaodong Yi, and Andreas Zell, “Multi-camera visual SLAM for autonomous navigation of micro aerial vehicles,” *Robotics and autonomous systems (Elsevier)*, vol. 93, pp. 116–134, 2017. (page 3).

[13] Bruno Ferrarini, Maria Waheed, Sania Waheed, Shoaib Ehsan, Michael Milford, and Klaus D McDonald-Maier, “Visual place recognition for aerial robotics: Exploring accuracy-computation trade-off for local image descriptors,” in *proceedings of the NASA/ESA conference on adaptive hardware and systems (AHS)*, Colchester, UK, July 2019, pp. 103–108. (page 3).

[14] E Ackerman. (2014) Dyson’s robot vacuum has 360-degree camera, tank treads, cyclone suction. [Online]. Available: http://spectrum.ieee.org/automaton/robotics/home-robots/dysonthe-360-eye-robot-vacuum (page 3).

[15] Mark Cummins and Paul Newman, “Probabilistic appearance based navigation and loop closing,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Rome, Italy, Aprils 2007, pp. 2042–2048. (pages 3 and 17).

[16] Paul Newman and Kin Ho, “SLAM-loop closing with visually salient features,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Barcelona, Spain, April 2005, pp. 635–642. (page 3).

[17] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. (page 3).

[18] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep learning*. MIT Press, Cambridge, MA, USA, 2016, vol. 1, no. 2. (page 3).

[19] Chingiz Kenshimov, Loukas Bampis, Beibut Amirgaliyev, Marat Arslanov, and Antonios Gasteratos, “Deep learning features exception for cross-season visual place recognition,” *Pattern recognition letters (Elsevier)*, vol. 100, pp. 124–130, 2017. (pages 3 and 83).

[20] Fabiola Maffra, Lucas Teixeira, Zetao Chen, and Margarita Chli, "Real-time wide-baseline place recognition using depth completion," *IEEE robotics and automation letters*, vol. 4, no. 2, pp. 1525–1532, 2019. (pages 3 and 84).

[21] Kin Leong Ho and Paul Newman, "Loop closure detection in SLAM by combining visual and spatial appearance," *Robotics and autonomous systems (Elsevier)*, vol. 54, no. 9, pp. 740–749, 2006. (page 4).

[22] Richard Hartley and Andrew Zisserman, *Multiple view geometry in computer vision.* Cambridge University Press, Cambridge, U.K., 2003. (page 3).

[23] Hisayoshi Sugiyama, Tetsuo Tsujioka, and Masashi Murata, "Collaborative movement of rescue robots for reliable and effective networking in disaster area," in *proceedings of the IEEE international conference on collaborative computing: networking, applications and worksharing (CollaborateCom)*, San Jose, CA, USA, December 2005, pp. 1–7. (page 3).

[24] Francesco Capezio, Fulvio Mastrogiovanni, Antonio Sgorbissa, and Renato Zaccaria, "Robot-assisted surveillance in large environments," *Journal of computing and information technology (CIT)*, vol. 17, no. 1, pp. 95–108, 2009. (page 3).

[25] Yvan Baudoin, Daniela Doroftei, Geert De Cubber, Sid Ahmed Berrabah, Carlos Pinzon, Fabrice Warlet, Jeremi Gancet, Elvina Motard, Michel Ilzkovitz, Lazaros Nalpantidis, *et al.*, "VIEW-FINDER: robotics assistance to fire-fighting services and crisis management," in *proceedings of the IEEE international workshop on safety, security and rescue robotics (SSRR)*, Denver, CO, USA, November 2009, pp. 1–6. (page 3).

[26] Ioannis Kostavelis, Lazaros Nalpantidis, Evangelos Boukas, Marcos Aviles Rodrigalvarez, Ioannis Stamoulias, George Lentaris, Dionysios Diamantopoulos, Kostas Siozios, Dimitrios Soudris, and Antonios Gasteratos, "SPARTAN: Developing a vision system for future autonomous space exploration robots," *Journal of field robotics (Wiley)*, vol. 31, no. 1, pp. 107–140, 2014. (page 3).

[27] Evangelos Boukas, Antonios Gasteratos, and Gianfranco Visentin, "Introducing a globally consistent orbital-based localization system," *Journal of field robotics (Wiley)*, vol. 35, no. 2, pp. 275–298, 2018. (page 3).

[28] Christopher J Cannell and Daniel J Stilwell, "A comparison of two approaches for adaptive sampling of environmental processes using autonomous underwater vehicles," in *proceedings of the IEEE OCEANS*, Washington, DC, USA, September 2005, pp. 1514–1521. (page 3).

[29] Shujing Zhang, Bo He, Rui Nian, Yan Liang, and Tianhong Yan, "SLAM and a novel loop closure detection for autonomous underwater vehicles," in *proceedings of the IEEE OCEANS*, San Diego, CA, USA, September 2013, pp. 1–4. (page 3).

[30] Min Jiang, Sanming Song, J Michael Herrmann, Ji-Hong Li, Yiping Li, Zhiqiang Hu, Zhigang Li, Jian Liu, Shuo Li, and Xisheng Feng, "Underwater loop-closure detection for mechanical scanning imaging sonar by filtering the similarity matrix with probability hypothesis density filter," *IEEE access*, vol. 7, pp. 166 614–166 628, 2019. (page 3).

[31] Naveed Muhammad, Juan Francisco Fuentes-Perez, Jeffrey A Tuhtan, Gert Toming, Mark Musall, and Maarja Kruusmaa, "Map-based localization and loop-closure detection from a moving underwater platform using flow features," *Autonomous robots (Springer)*, vol. 43, no. 6, pp. 1419–1434, 2019. (page 3).

[32] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, José Neira, Ian Reid, and John J Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE transactions on robotics*, vol. 32, no. 6, pp. 1309–1332, 2016. (page 5).

[33] Sebastian Thrun, Wolfram Burgard, and Dieter Fox, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Autonomous robots (Springer)*, vol. 5, no. 3, pp. 253–271, 1998. (page 5).

[34] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015. (page 5).

[35] Brian Williams, Georg Klein, and Ian Reid, "Automatic relocalization and loop closing for real-time monocular SLAM," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 9, pp. 1699–1712, 2011. (page 5).

[36] Jörg Röwekämper, Christoph Sprunk, Gian Diego Tipaldi, Cyrill Stachniss, Patrick Pfaff, and Wolfram Burgard, "On the position accuracy of mobile robot localization based on particle filters combined with scan matching," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Vilamoura-Algarve, Portugal, October 2012, pp. 3158–3164. (page 6).

[37] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007. (page 6).

[38] Edward C Tolman, "Cognitive maps in rats and men," *Psychological review*, vol. 55, no. 4, p. 189, 1948. (page 6).

[39] Felix Strumwasser, “Long-term recording from single neurons in brain of unrestrained mammals,” *Science (AAAS)*, vol. 127, no. 3296, pp. 469–470, 1958. (page 6).

[40] John O’Keefe and Jonathan Dostrovsky, “The hippocampus as a spatial map: preliminary evidence from unit activity in the freely-moving rat,” *Brain research (Elsevier)*, vol. 34, no. 1, pp. 171–175, 1971. (page 6).

[41] Benjamin Kuipers and Yung-Tai Byun, “A robust qualitative method for spatial learning in unknown environments,” in *proceedings of the AAAI conference on artificial intelligence*, Saint Paul, Minnesota, August 1988, pp. 774–779. (page 6).

[42] Matthias O Franz, Bernhard Schölkopf, Hanspeter A Mallot, and Heinrich H Bülthoff, “Learning view graphs for robot navigation,” *Autonomous robots (Springer)*, vol. 5, no. 1, pp. 111–125, 1998. (page 6).

[43] Howie Choset and Keiji Nagatani, “Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization,” *IEEE transactions on robotics and automation*, vol. 17, no. 2, pp. 125–137, 2001. (page 6).

[44] Francesco Savelli and Benjamin Kuipers, “Loop-closing and planarity in topological map-building,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Sendai, Japan, September 2004, pp. 1511–1517. (page 6).

[45] Ananth Ranganathan, Emanuele Menegatti, and Frank Dellaert, “Bayesian inference in the space of topological maps,” *IEEE transactions on robotics*, vol. 22, no. 1, pp. 92–107, 2006. (page 6).

[46] Ananth Ranganathan and Frank Dellaert, “Online probabilistic topological mapping,” *The international journal of robotics research (SAGE)*, vol. 30, no. 6, pp. 755–771, 2011. (page 6).

[47] Ethan Eade and Tom Drummond, “Unified loop closing and recovery for real time monocular SLAM,” in *proceedings of the British machine vision conference (BMVC)*, vol. 13, Leeds, UK, September 2008, p. 136. (pages 6 and 85).

[48] Benjamin Kuipers, “Modeling spatial knowledge,” *Cognitive Science (Elsevier)*, vol. 2, no. 2, pp. 129–153, 1978. (page 6).

[49] Benjamin Kuipers and Yung-Tai Byun, “A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations,” *Robotics and autonomous systems (Elsevier)*, vol. 8, no. 1-2, pp. 47–63, 1991. (page 6).

[50] Benjamin Kuipers, Joseph Modayil, Patrick Beeson, Matt MacMahon, and Francesco Savelli, "Local metrical and global topological maps in the hybrid spatial semantic hierarchy," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, vol. 5, New Orleans, LA, USA, April 2004, pp. 4845–4851. (page 6).

[51] Jose-Luis Blanco, Juan-Antonio Fernández-Madrigal, and Javier Gonzalez, "Toward a unified Bayesian approach to hybrid metric-topological SLAM," *IEEE transactions on robotics*, vol. 24, no. 2, pp. 259–270, 2008. (pages 6 and 17).

[52] Johann Borenstein and Liqiang Feng, "Measurement and correction of systematic odometry errors in mobile robots," *IEEE transactions on robotics and automation*, vol. 12, no. 6, pp. 869–880, 1996. (page 7).

[53] J-S Gutmann and Kurt Konolige, "Incremental mapping of large cyclic environments," in *proceedings of the IEEE international symposium on computational intelligence in robotics and automation*, Monterey, CA, USA, November 1999, pp. 318–325. (page 7).

[54] Michael Bosse and Robert Zlot, "Keypoint design and evaluation for place recognition in 2D LiDAR maps," *Robotics and autonomous systems (Elsevier)*, vol. 57, no. 12, pp. 1211–1224, 2009. (page 7).

[55] Martin Magnusson, Henrik Andreasson, Andreas Nuchter, and Achim J Lilienthal, "Appearance-based loop detection from 3D laser data using the normal distributions transform," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Kobe, Japan, May 2009, pp. 23–28. (page 7).

[56] Bastian Steder, Michael Ruhnke, Slawomir Grzonka, and Wolfram Burgard, "Place recognition in 3D scans using a combination of bag of words and point feature based relative pose estimation," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, San Francisco, CA, USA, September 2011, pp. 1249–1255. (page 7).

[57] Michael Bosse and Robert Zlot, "Place recognition using keypoint voting in large 3D lidar datasets," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Karlsruhe, Germany, May 2013, pp. 2677–2684. (pages 7 and 86).

[58] D. Hahnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Las Vegas, NV, USA, October 2003, pp. 206–211. (page 7).

[59] Peter Biber, Tom Duckett, *et al.*, "Dynamic maps for long-term operation of mobile service robots," in *proceedings of the robotics: science and systems (RSS)*, Cambridge, MA, USA, June 2005, pp. 17–24. (page 7).

[60] Wolfram Burgard, Cyrill Stachniss, and Dirk Hähnel, "Mobile robot map learning from range data in dynamic environments," in *Autonomous navigation in dynamic Environments*. Springer-Verlag, Berlin, Germany, 2007, pp. 3–28. (page 7).

[61] Michael Bosse and Jonathan Roberts, "Histogram matching and global initialization for laser-only SLAM in large unstructured environments," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Rome, Italy, April 2007, pp. 4820–4826. (page 7).

[62] Daniele Cattaneo, Matteo Vaghi, Simone Fontana, Augusto Luis Ballardini, and Domenico Giorgio Sorrenti, "Global visual localization in LiDAR-maps through shared 2D-3D embedding space," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Paris, France, May 2020, pp. 4365–4371. (page 7).

[63] Andrew J Davison and David W. Murray, "Simultaneous localization and map-building using active vision," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 7, pp. 865–880, 2002. (page 7).

[64] Michael Milford and Gordon Wyeth, "Hippocampal models for simultaneous localisation and mapping on an autonomous robot," in *proceedings of the Australasian conference on robotics and automation (ACRA)*, Brisbane, Australia, May 2003, pp. 1–10. (page 7).

[65] Michael Bosse, Paul Newman, John Leonard, Martin Soika, Wendelin Feiten, and Seth Teller, "An Atlas framework for scalable mapping," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Taipei, Taiwan, September 2003, pp. 1899–1906. (page 7).

[66] Michael J Milford, Gordon F Wyeth, and David Prasser, "RatSLAM: A hippocampal model for simultaneous localization and mapping," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, New Orleans, LA, USA, April 2004, pp. 403–408. (pages 7 and 87).

[67] Paul Newman, David Cole, and Kin Ho, "Outdoor SLAM using visual appearance and laser ranging," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Orlando, FL, USA, May 2006, pp. 1180–1187. (pages 7, 15, and 16).

[68] Friedrich Fraundorfer, Christopher Engels, and David Nistér, "Topological mapping, localization and navigation using image collections," in *proceedings of the*

IEEE/RSJ international conference on intelligent robots and systems (IROS), San Diego, CA, USA, October 2007, pp. 3872–3877. (page 7).

[69] Pedro Piniés and Juan D Tardós, "Large-scale SLAM building conditionally independent local maps: Application to monocular vision," *IEEE transactions on robotics*, vol. 24, no. 5, pp. 1094–1106, 2008. (page 7).

[70] Wei Tan, Haomin Liu, Zilong Dong, Guofeng Zhang, and Hujun Bao, "Robust monocular SLAM in dynamic environments," in *proceedings of the IEEE international symposium on mixed and augmented reality (ISMAR)*, Adelaide, SA, Australia, October 2013, pp. 209–218. (pages 7 and 87).

[71] Stephen Se, David Lowe, and Jim Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *The international journal of robotics research (SAGE)*, vol. 21, no. 8, pp. 735–758, 2002. (page 7).

[72] Kurt Konolige and Motilal Agrawal, "FrameSLAM: From bundle adjustment to real-time visual mapping," *IEEE transactions on robotics*, vol. 24, no. 5, pp. 1066–1077, 2008. (page 7).

[73] Lina M Paz, Pedro Piniés, Juan D Tardós, and José Neira, "Large-scale 6-DOF SLAM with stereo-in-hand," *IEEE transactions on robotics*, vol. 24, no. 5, pp. 946–957, 2008. (page 7).

[74] Paul Newman, Gabe Sibley, Mike Smith, Mark Cummins, Alastair Harrison, Chris Mei, Ingmar Posner, Robbie Shade, Derik Schroeter, Liz Murphy, *et al.*, "Navigating, recognizing and describing urban spaces with vision and lasers," *The international journal of robotics research (SAGE)*, vol. 28, no. 11-12, pp. 1406–1433, 2009. (pages 7 and 18).

[75] Gabe Sibley, Christopher Mei, Ian Reid, and Paul Newman, "Vast-scale outdoor navigation using adaptive relative bundle adjustment," *The international journal of robotics research (SAGE)*, vol. 29, no. 8, pp. 958–980, 2010. (pages 7 and 14).

[76] Lazaros Nalpantidis, Georgios Ch Sirakoulis, and Antonios Gasteratos, "Non-probabilistic cellular automata-enhanced stereo vision simultaneous localization and mapping," *Measurement science and technology (IOPscience)*, vol. 22, no. 11, p. 114027, 2011. (page 7).

[77] César Cadena, Dorian Gálvez-López, Juan D Tardós, and José Neira, "Robust place recognition with stereo sequences," *IEEE transactions on robotics*, vol. 28, no. 4, pp. 871–885, 2012. (page 7).

[78] José A Castellanos, José Neira, and Juan D Tardós, "Multisensor fusion for simultaneous localization and map building," *IEEE transactions on robotics and automation*, vol. 17, no. 6, pp. 908–914, 2001. (page 7).

[79] Rohan Paul and Paul Newman, “FAB-MAP 3D: Topological mapping with spatial and visual appearance,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Anchorage, AK, USA, May 2010, pp. 2649–2656. (page 7).

[80] Edward Pepperell, Peter I Corke, and Michael J Milford, “All-environment visual place recognition with SMART,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, May 2014, pp. 1612–1618. (pages 7 and 81).

[81] Stefan Leutenegger, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale, “Keyframe-based visual-inertial odometry using nonlinear optimization,” *The international journal of robotics research (SAGE)*, vol. 34, no. 3, pp. 314–334, 2015. (page 7).

[82] Stephen Hausler, Adam Jacobson, and Michael Milford, “Multi-process fusion: Visual place recognition using multiple image processing methods,” *IEEE robotics and automation letters*, vol. 4, no. 2, pp. 1924–1931, 2019. (page 7).

[83] Hernán Badino, Daniel Huber, and Takeo Kanade, “Real-time topometric localization,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Saint Paul, MN, USA, May 2012, pp. 1635–1642. (pages 7 and 9).

[84] Mark Cummins and Paul Newman, “Appearance-only SLAM at large scale with FAB-MAP 2.0,” *The international journal of robotics research (SAGE)*, vol. 30, no. 9, pp. 1100–1123, 2011. (pages 7, 17, 27, and 87).

[85] Bernt Schiele and James L Crowley, “Object recognition using multidimensional receptive field histograms,” in *proceedings of the European conference on computer vision (ECCV)*, Cambridge, UK, April 1996, pp. 610–619. (page 9).

[86] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, “SURF: speeded up robust features,” in *proceedings of the European conference on computer vision (ECCV)*, Graz, Austria, May 2006, pp. 404–417. (pages 9, 10, 40, 43, 47, 50, 57, 69, and 72).

[87] Mary C Potter, “Meaning in visual search,” *Science (AAAS)*, vol. 187, no. 4180, pp. 965–966, 1975. (page 8).

[88] Irving Biederman, “Aspects and extensions of a theory of human image understanding,” in *Computational processes in human vision: An interdisciplinary perspective*. Ablex Publishing Corporation: Norwood, New Jersey, 1988, pp. 370–428. (page 8).

[89] Aude Oliva and Antonio Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International journal of computer vision (Springer)*, vol. 42, no. 3, pp. 145–175, 2001. (page 9).

[90] Antonio Torralba, Kevin P Murphy, William T Freeman, Mark A Rubin, *et al.*, “Context-based vision system for place and object recognition,” in *proceedings of the IEEE international conference on computer vision (ICCV)*, vol. 3, Nice, France, October 2003, pp. 273–280. (page 9).

[91] Aude Oliva and Antonio Torralba, “Building the gist of a scene: The role of global image features in recognition,” *Progress in brain research (Elsevier)*, vol. 155, pp. 23–36, 2006. (page 9).

[92] Ana C Murillo and Jana Kosecka, “Experiments in place recognition using gist panoramas,” in *proceedings of the international conference on computer vision workshops (ICCV)*, Kyoto, Japan, September 2009, pp. 2196–2203. (page 9).

[93] Gautam Singh and Jana Kosecka, “Visual loop closing using gist descriptors in manhattan world,” in *proceedings of the IEEE international conference on robotics and automation workshop (ICRA)*, Anchorage, Alaska, May 2010, pp. 4042–4047. (page 9).

[94] Yang Liu and Hong Zhang, “Visual loop closure detection with a compact image descriptor,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Vilamoura-Algarve, Portugal, October 2012, pp. 1051–1056. (pages 9 and 86).

[95] S. M. A. M. Kazmi and B. Mertsching, “Detecting the expectancy of a place using nearby context for appearance-based mapping,” *IEEE transactions on robotics*, vol. 35, no. 6, pp. 1352–1366, 2019. (pages 9, 17, 27, 61, 82, and 84).

[96] Niko Sünderhauf and Peter Protzel, “BRIEF-gist closing the loop by simple means,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, San Francisco, CA, USA, September 2011, pp. 1234–1241. (page 9).

[97] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua, “BRIEF: binary robust independent elementary features,” in *proceedings of the European conference on computer vision (ECCV)*, Heraklion, Crete, Greece, September 2010, pp. 778–792. (page 9).

[98] Roberto Arroyo, Pablo F Alcantarilla, Luis M Bergasa, J Javier Yebes, and Sebastián Bronte, “Fast and effective visual place recognition using binary codes

and disparity information," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Chicago, IL, USA, September 2014, pp. 3089–3094. (pages 9, 24, and 81).

[99] Xin Yang and Kwang-Ting Cheng, "LDB: An ultra-fast feature for scalable augmented reality on mobile devices," in *proceedings of the IEEE international symposium on mixed and augmented reality (ISMAR)*, Atlanta, GA, USA, November 2012, pp. 49–57. (page 9).

[100] Iwan Ulrich and Illah Nourbakhsh, "Appearance-based place recognition for topological localization," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, San Francisco, CA, USA, April 2000, pp. 1023–1029. (pages 9 and 15).

[101] Anat Levin and Richard Szeliski, "Visual odometry and map correlation," in *proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)*, Washington, DC, USA, June 2004, pp. I–I. (page 9).

[102] L Maohai, S Lining, H Qingcheng, C Zesu, and P Songhao, "Robust omnidirectional vision based mobile robot hierarchical localization and autonomous navigation," *Information technology journal (Science Alert)*, vol. 10, no. 1, pp. 29–39, 2011. (pages 9 and 85).

[103] William T Freeman and Michal Roth, "Orientation histograms for hand gesture recognition," in *proceedings of the international workshop on automatic face and gesture recognition*, Zurich, Switzerland, June 1995, pp. 296–301. (page 9).

[104] Jana Kosecka, Liang Zhou, Philip Barber, and Zoran Duric, "Qualitative image based localization in indoors environments," in *proceedings of the IEEE computer society conference on computer vision and pattern recognition (CVPR)*, Madison, WI, USA, June 2003, pp. II–II. (page 9).

[105] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *proceedings of the IEEE computer society conference on computer vision and rattern recognition (CVPR)*, San Diego, CA, USA, June 2005, pp. 886–893. (page 9).

[106] Sayem Mohammad Siam and Hong Zhang, "Fast-SeqSLAM: A fast appearance based place recognition algorithm," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Singapore, June 2017, pp. 5702–5708. (pages 9 and 87).

[107] Jie Luo, Andrzej Pronobis, Barbara Caputo, and Patric Jensfelt, "Incremental learning for place recognition in dynamic environments," in *proceedings of the*

IEEE/RSJ international conference on intelligent robots and systems (IROS), San Diego, CA, USA, October 2007, pp. 721–728. (page 9).

[108] Anna Bosch, Andrew Zisserman, and Xavier Munoz, “Representing shape with a spatial pyramid kernel,” in *proceedings of the ACM international conference on image and video retrieval (CIVR)*, Amsterdam, the Netherlands, July 2007, pp. 401–408. (page 9).

[109] Wei-Chen Chiu and Mario Fritz, “See the difference: Direct pre-image reconstruction and pose estimation by differentiating HOG,” in *proceedings of the IEEE international conference on computer vision (ICCV)*, Santiago, Chile, December 2015, pp. 468–476. (page 9).

[110] Michael J Milford and Gordon F Wyeth, “SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Saint Paul, MN, USA, May 2012, pp. 1643–1649. (pages 9, 14, 17, 26, 40, 44, 81, and 86).

[111] Gregory Dudek and Deeptiman Jugessur, “Robust place recognition using local appearance based methods,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, San Francisco, CA, USA, April 2000, pp. 1030–1035. (pages 9 and 16).

[112] David G Lowe, “Distinctive image features from scale-invariant keypoints,” *International journal of computer vision (Springer)*, vol. 60, no. 2, pp. 91–110, 2004. (pages 10, 44, 58, and 59).

[113] Motilal Agrawal, Kurt Konolige, and Morten Rufus Blas, “CenSurE: Center surround extremas for realtime feature detection and matching,” in *proceedings of the European conference on computer vision (ECCV)*, Marseille, France, October 2008, pp. 102–115. (page 10).

[114] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison, “KAZE features,” in *proceedings of the European conference on computer vision (ECCV)*, Florence, Italy, October 2012, pp. 214–227. (page 10).

[115] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski, “ORB: An efficient alternative to SIFT or SURF,” in *proceedings of the international conference on computer vision (ICCV)*, Barcelona, Spain, November 2011, pp. 2564–2571. (page 10).

[116] Stefan Leutenegger, Margarita Chli, and Roland Y Siegwart, “BRISK: Binary robust invariant scalable keypoints,” in *proceedings of the international conference on computer vision (ICCV)*, Barcelona, Spain, November 2011, pp. 2548–2555. (page 10).

[117] Alexandre Alahi, Raphael Ortiz, and Pierre Vandergheynst, "FREAK: Fast retina keypoint," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Providence, RI, USA, June 2012, pp. 510–517. (page 10).

[118] Pablo F Alcantarilla and T Solutions, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1281–1298, 2011. (page 10).

[119] Xiang Sean Zhou and Thomas S Huang, "Edge-based structural features for content-based image retrieval," *Pattern recognition letters (Elsevier)*, vol. 22, no. 5, pp. 457–468, 2001. (page 10).

[120] Joan P Company-Corcoles, Emilio Garcia-Fidalgo, and Alberto Ortiz, "Towards robust loop closure detection in weakly textured environments using points and lines," in *proceedings of the IEEE international conference on emerging technologies and factory automation (ETFA)*, Vienna, Austria, September 2020, pp. 1313–1316. (page 10).

[121] Grant Schindler, Matthew Brown, and Richard Szeliski, "City-scale location recognition," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Minneapolis, MN, USA, June 2007, pp. 1–7. (pages 10 and 87).

[122] Hong Zhang, "BoRF: Loop-closure detection with scale invariant visual features," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Shanghai, China, May 2011, pp. 3125–3130. (pages 10 and 14).

[123] Panu Turcot and David G Lowe, "Better matching with fewer features: The selection of useful features in large database recognition problems," in *proceedings of the IEEE international conference on computer vision workshops (ICCV)*, Kyoto, Japan, September 2009, pp. 2109–2116. (page 10).

[124] Simon Lynen, Michael Bosse, Paul Furgale, and Roland Siegwart, "Placeless place-recognition," in *proceedings of the IEEE international conference on 3D vision*, Tokyo, Japan, December 2014, pp. 303–310. (pages 10, 59, and 87).

[125] Mathias Gehrig, Elena Stumm, Timo Hinzmann, and Roland Siegwart, "Visual place recognition with probabilistic voting," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Singapore, June 2017, pp. 3192–3199. (pages 10, 16, 26, and 27).

[126] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, *et al.*, *Modern information retrieval.* ACM press New York, 1999, vol. 463. (page 10).

[127] Josef Sivic and Andrew Zisserman, “Video Google: A text retrieval approach to object matching in videos,” in *proceedings of the IEEE international conference on computer vision (ICCV)*, Nice, France, October 2003, p. 1470. (page 10).

[128] James MacQueen *et al.*, “Some methods for classification and analysis of multivariate observations,” in *proceedings of the Berkeley symposium on mathematical statistics and probability*, Berkeley, CA, USA, January 1967, pp. 281–297. (page 10).

[129] Mark Cummins and Paul Newman, “FAB-MAP: probabilistic localization and mapping in the space of appearance,” *The international journal of robotics research (SAGE)*, vol. 27, no. 6, pp. 647–665, 2008. (pages 10, 14, 15, 17, 18, 26, 47, 49, 50, 77, and 87).

[130] Hemanth Korrapati and Youcef Mezouar, “Vision-based sparse topological mapping,” *Robotics and autonomous systems (Elsevier)*, vol. 62, no. 9, pp. 1259–1270, 2014. (page 10).

[131] Dorian Gálvez-López and Juan D Tardos, “Bags of binary words for fast place recognition in image sequences,” *IEEE transactions on robotics*, vol. 28, no. 5, pp. 1188–1197, 2012. (pages 10, 15, 27, 35, and 86).

[132] Thomas Whelan, Michael Kaess, Hordur Johannsson, Maurice Fallon, John J Leonard, and John McDonald, “Real-time large-scale dense RGB-D SLAM with volumetric fusion,” *The international journal of robotics research (SAGE)*, vol. 34, no. 4-5, pp. 598–626, 2014. (page 10).

[133] Loukas Bampis, Angelos Amanatiadis, and Antonios Gasteratos, “Encoding the description of image sequences: A two-layered pipeline for loop closure detection,” in *proceedings of the IEEE international conference on intelligent robots and systems (IROS)*, Daejeon, Korea (South), October 2016, pp. 4530–4536. (pages 10 and 17).

[134] Karen Sparck Jones, “A statistical interpretation of term specificity and its application in retrieval,” *Journal of documentation (Emerald Publishing)*, vol. 28, no. 1, pp. 11–21, 1972. (page 10).

[135] Djoerd Hiemstra, “A probabilistic justification for using tf × idf term weighting in information retrieval,” *International journal on digital libraries (Springer)*, vol. 3, no. 2, pp. 131–139, 2000. (page 11).

[136] Florent Perronnin and Christopher Dance, “Fisher kernels on visual vocabularies for image categorization,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Minneapolis, MN, USA, June 2007, pp. 1–8. (page 11).

[137] Relja Arandjelovic and Andrew Zisserman, “All about VLAD,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Portland, Oregon, June 2013, pp. 1578–1585. (page 11).

[138] Yasir Latif, Guoquan Huang, John J Leonard, and José Neira, “An online sparsity-cognizant loop-closure algorithm for visual navigation,” in *proceedings of the robotics: science and systems (RSS)*, Rome, Italy, July 2014. (page 11).

[139] David Filliat, “A visual bag of words method for interactive qualitative localization and mapping,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Rome, Italy, April 2007, pp. 3921–3926. (page 11).

[140] Adrien Angeli, David Filliat, Stéphane Doncieux, and Jean-Arcady Meyer, “Fast and incremental method for loop-closure detection using bags of visual words,” *IEEE transactions on robotics*, vol. 24, no. 5, pp. 1027–1037, 2008. (pages 11, 17, 25, 47, 50, 68, 69, 73, and 79).

[141] Tudor Nicosevici and Rafael Garcia, “On-line visual vocabularies for robot navigation and mapping,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, St. Louis, MO, USA, October 2009, pp. 205–212. (page 11).

[142] Yogesh Girdhar and Gregory Dudek, “Online visual vocabularies,” in *proceedings of the IEEE Canadian conference on computer and robot vision*, St. John’s, NL, Canada, May 2011, pp. 191–196. (page 11).

[143] Aram Kawewong, Noppharit Tongprasit, Sirinart Tangruamsub, and Osamu Hasegawa, “Online and incremental appearance-based SLAM in highly dynamic environments,” *The international journal of robotics research (SAGE)*, vol. 30, no. 1, pp. 33–55, 2011. (page 11).

[144] Tudor Nicosevici and Rafael Garcia, “Automatic visual bag-of-words for online robot navigation and mapping,” *IEEE transactions on robotics*, vol. 28, no. 4, pp. 886–898, 2012. (page 11).

[145] Mathieu Labbe and Francois Michaud, “Appearance-based loop closure detection for online large-scale and long-term operation,” *IEEE transactions on robotics*, vol. 29, no. 3, pp. 734–745, 2013. (pages 11, 17, 61, and 86).

[146] Emilio Garcia-Fidalgo and Alberto Ortiz, “On the use of binary feature descriptors for loop closure detection,” in *proceedings of the IEEE emerging technology and factory automation (ETFA)*, Barcelona, Spain, September 2014, pp. 1–8. (page 11).

[147] Sheraz Khan and Dirk Wollherr, "IBuILD: Incremental bag of binary words for appearance based loop closure detection," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA, May 2015, pp. 5441–5447. (pages 11, 16, 17, and 27).

[148] Guangcong Zhang, Mason J Lilly, and Patricio A Vela, "Learning binary features online from motion dynamics for incremental loop-closure detection and place recognition," in *proceedings of the international conference on robotics and automation (ICRA)*, Stockholm, Sweden, May 2016, pp. 765–772. (page 11).

[149] Emilio Garcia-Fidalgo and Alberto Ortiz, "Hierarchical place recognition for topological mapping," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1061–1074, 2017. (pages 11, 61, 85, and 86).

[150] ——, "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. (pages 11 and 26).

[151] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989. (page 11).

[152] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," *proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. (page 11).

[153] Zhenguang Liu, Luming Zhang, Qi Liu, Yifang Yin, Li Cheng, and Roger Zimmermann, "Fusion of magnetic and visual sensors for indoor localization: Infrastructure-free and more effective," *IEEE transactions on multimedia*, vol. 19, no. 4, pp. 874–888, 2016. (page 11).

[154] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012. (pages 11 and 12).

[155] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *proceedings of the international conference on learning representations workshop (ICLR)*, Banff, Canada, April 2014. (page 11).

[156] Will Maddern, Alex Stewart, Colin McManus, Ben Upcroft, Winston Churchill, and Paul Newman, "Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles," in

proceedings of the visual place recognition in changing environments workshop, IEEE international conference on robotics and automation (ICRA), Hong Kong, China, May 2014, p. 3. (pages 11, 82, and 83).

[157] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision (Springer)*, vol. 115, no. 3, pp. 211–252, 2015. (page 11).

[158] Artem Babenko and Victor Lempitsky, "Aggregating deep convolutional features for image retrieval," *arXiv preprint arXiv:1510.07493*, 2015. (page 11).

[159] Filip Radenović, Giorgos Tolias, and Ondřej Chum, "CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples," in *proceedings of the European conference on computer vision (ECCV)*, Amsterdam, the Netherlands, October 2016, pp. 3–20. (pages 11 and 88).

[160] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus, "End-to-end learning of deep visual representations for image retrieval," *International journal of computer vision (Springer)*, vol. 124, no. 2, pp. 237–254, 2017. (page 11).

[161] P Rolet, M Sebag, and O Teytaud, "Integrated recognition, localization and detection using convolutional networks," in *proceedings of the European conference on machine learning*, Bristol, UK, September 2012, pp. 1255–1263. (pages 11 and 12).

[162] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, June 2016, pp. 5297–5307. (pages 11 and 12).

[163] Jun Yu, Chaoyang Zhu, Jian Zhang, Qingming Huang, and Dacheng Tao, "Spatial pyramid-enhanced NetVLAD with weighted triplet loss for place recognition," *IEEE transactions on neural networks and learning systems*, vol. 31, no. 2, pp. 661–674, 2019. (pages 11 and 12).

[164] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky, "Neural codes for image retrieval," in *proceedings of the European conference on computer vision (ECCV)*, Zurich, Switzerland, September 2014, pp. 584–599. (pages 12 and 13).

[165] Niko Sünderhauf, Sareh Shirazi, Adam Jacobson, Feras Dayoub, Edward Pepperell, Ben Upcroft, and Michael Milford, "Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free," in *proceedings of the*

robotics: science and systems (RSS), Rome, Italy, July 2015, pp. 1–10. (pages 12, 83, and 84).

[166] Aravindh Mahendran and Andrea Vedaldi, "Understanding deep image representations by inverting them," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Boston, MA, USA, June 2015, pp. 5188–5196. (page 12).

[167] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal Fua, "LIFT: Learned invariant feature transform," in *proceedings of the European conference on computer vision (ECCV)*, Amsterdam, the Netherlands, October 2016, pp. 467–483. (page 12).

[168] Silvia Cascianelli, Gabriele Costante, Enrico Bellocchio, Paolo Valigi, Mario L Fravolini, and Thomas A Ciarfuglia, "Robust visual semi-semantic loop closure detection by a covisibility graph and CNN features," *Robotics and autonomous systems (Elsevier)*, vol. 92, pp. 53–65, 2017. (pages 12 and 84).

[169] Giorgos Tolias, Ronan Sicre, and Hervé Jégou, "Particular object retrieval with integral max-pooling of cnn activations," *arXiv preprint arXiv:1511.05879*, 2015. (pages 12 and 13).

[170] Ali S Razavian, Josephine Sullivan, Stefan Carlsson, and Atsuto Maki, "Visual instance retrieval with deep convolutional networks," *ITE transactions on media technology and applications*, vol. 4, no. 3, pp. 251–258, 2016. (page 12).

[171] Peer Neubert and Peter Protzel, "Beyond holistic descriptors, keypoints, and fixed patches: Multiscale superpixel grids for place recognition in changing environments," *IEEE robotics and automation letters*, vol. 1, no. 1, pp. 484–491, 2016. (pages 12, 13, 82, and 84).

[172] Giorgos Tolias, Yannis Avrithis, and Hervé Jégou, "Image search with selective match kernels: aggregation across single and multiple images," *International journal of computer vision (Springer)*, vol. 116, no. 3, pp. 247–261, 2016. (page 12).

[173] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han, "Large-scale image retrieval with attentive deep local features," in *proceedings of the IEEE international conference on computer vision (ICCV)*, Venice, Italy, October 2017, pp. 3456–3465. (pages 12 and 13).

[174] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *proceedings of the IEEE computer society conference on computer vision and pattern recognition*

workshop (CVPR), Salt Lake City, UT, USA, June 2018, pp. 224–236. (pages 12, 13, and 82).

[175] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk, "Key.net: Keypoint detection by handcrafted and learned CNN filters," in *proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, Seoul, Korea (South), October 2019, pp. 5836–5844. (page 12).

[176] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler, "D2-net: A trainable cnn for joint description and detection of local features," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 8092–8101. (pages 12, 13, and 82).

[177] Zetao Chen, Obadiah Lam, Adam Jacobson, and Michael Milford, "Convolutional neural network-based place recognition," *arXiv preprint arXiv:1411.1509*, 2014. (pages 12 and 84).

[178] Niko Sünderhauf, Sareh Shirazi, Feras Dayoub, Ben Upcroft, and Michael Milford, "On the performance of convnet features for place recognition," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Hamburg, Germany, September 2015, pp. 429–4304. (pages 12 and 84).

[179] Thomas Whelan, Michael Kaess, John J Leonard, and John McDonald, "Deformation-based loop closure for large scale dense RGB-D SLAM," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Tokyo, Japan, November 2013, pp. 548–555. (pages 12 and 85).

[180] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *proceedings of the IEEE conference on computer vision and pattern recognition workshop (CVPR)*, Columbus, OH, USA, June 2014, pp. 806–813. (page 12).

[181] Ross Finman, Liam Paull, and John J Leonard, "Toward object-based place recognition in dense RGB-D maps," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, vol. 76, Seattle, WA, USA, May 2015. (pages 12 and 18).

[182] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully convolutional networks for semantic segmentation," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Boston, MA, USA, June 2015, pp. 3431–3440. (page 12).

[183] Erik Stenborg, Carl Toft, and Lars Hammarstrand, "Long-term visual localization using semantically segmented images," in *proceedings of the IEEE international*

conference on robotics and automation (ICRA), Brisbane, Australia, May 2018, pp. 6484–6490. (pages 12, 82, and 86).

[184] Matthew D Zeiler and Rob Fergus, "Visualizing and understanding convolutional networks," in *proceedings of the European conference on computer vision (ECCV)*, Zurich, Switzerland, September 2014, pp. 818–833. (page 12).

[185] Ioannis Kansizoglou, Loukas Bampis, and Antonios Gasteratos, "Deep feature space: A geometrical perspective," *IEEE transactions on pattern analysis and machine intelligence*, 2021. (page 12).

[186] Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *proceedings of the IEEE international conference on computer vision (ICCV)*, Santiago, Chile, December 2015, pp. 118–126. (page 12).

[187] Yifan Xia, Jie Li, Lin Qi, and Hao Fan, "Loop closure detection for visual SLAM using PCANet features," in *proceedings of the IEEE international joint conference on neural networks (IJCNN)*, Vancouver, BC, Canada, July 2016, pp. 2274–2281. (page 12).

[188] Sourav Garg, Niko Suenderhauf, and Michael Milford, "Don't look back: Robustifying place categorization for viewpoint-and condition-invariant place recognition," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Brisbane, Australia, May 2018, pp. 3645–3652. (pages 12 and 84).

[189] Torsten Sattler, Will Maddern, Carl Toft, Akihiko Torii, Lars Hammarstrand, Erik Stenborg, Daniel Safari, Masatoshi Okutomi, Marc Pollefeys, Josef Sivic, *et al.*, "Benchmarking 6dof outdoor visual localization in changing conditions," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Salt Lake City, UT, USA, June 2018, pp. 8601–8610. (pages 12 and 82).

[190] Shuo Wang, Xudong Lv, Xiaomin Liu, and Dong Ye, "Compressed holistic convnet representations for detecting loop closures in dynamic environments," *IEEE Access*, vol. 8, pp. 60 552–60 574, 2020. (pages 12 and 88).

[191] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014. (page 12).

[192] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *proceedings of the IEEE conference on computer*

vision and pattern recognition (CVPR), Las Vegas, NV, USA, June 2016, pp. 770–778. (page 12).

[193] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI conference on artificial intelligence*, San Francisco, CA, USA, February 2017. (page 12).

[194] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, HI, USA, July 2017, pp. 4700–4708. (page 12).

[195] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017. (page 12).

[196] JH Oh, JD Jeon, and BH Lee, "Place recognition for visual loop-closures using similarities of object graphs," *IET electronics letters*, vol. 51, no. 1, pp. 44–46, 2014. (page 13).

[197] Carl Toft, Carl Olsson, and Fredrik Kahl, "Long-term 3D localization and pose from semantic labellings," in *proceedings of the IEEE international conference on computer vision workshop (ICCV)*, Venice, Italy, October 2017, pp. 650–659. (page 13).

[198] Xin Yu, Sagar Chaturvedi, Chen Feng, Yuichi Taguchi, Teng-Yok Lee, Clinton Fernandes, and Srikumar Ramalingam, "VLASE: Vehicle localization by aggregating semantic edges," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Madrid, Spain, October 2018, pp. 3196–3203. (page 13).

[199] Abel Gawel, Carlo Del Don, Roland Siegwart, Juan Nieto, and Cesar Cadena, "X-VIEW: Graph-based semantic multi-view localization," *IEEE robotics and automation letters*, vol. 3, no. 3, pp. 1687–1694, 2018. (pages 13, 82, and 84).

[200] Ioannis Tsampikos Papapetros, Vasiliki Balaska, and Antonios Gasteratos, "Multilayer map: Augmenting semantic visual memory," in *proceedings of the IEEE international conference on unmanned aircraft systems (ICUAS)*, Athens, Greece, June 2020, pp. 1206–1212. (pages 13 and 84).

[201] Zhiding Yu, Chen Feng, Ming-Yu Liu, and Srikumar Ramalingam, "Casenet: Deep category-aware semantic edge detection," in *proceedings of the IEEE*

conference on computer vision and pattern recognition (CVPR), Honolulu, HI, USA, July 2017, pp. 5964–5973. (page 13).

[202] Zetao Chen, Fabiola Maffra, Inkyu Sa, and Margarita Chli, "Only look once, mining distinctive landmarks from convnet for visual place recognition," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Vancouver, Canada, September 2017, pp. 9–16. (pages 13, 82, 84, and 88).

[203] Zetao Chen, Lingqiao Liu, Inkyu Sa, Zongyuan Ge, and Margarita Chli, "Learning context flexible attention model for long-term visual place recognition," *IEEE robotics and automation letters*, vol. 3, no. 4, pp. 4015–4022, 2018. (page 13).

[204] Luis G Camara and Libor Přeučil, "Spatio-semantic ConvNet-based visual place recognition," in *proceedings of the European conference on mobile robots (ECMR)*, Prague, Czech Republic, September 2019, pp. 1–8. (page 13).

[205] Ahmad Khaliq, Shoaib Ehsan, Zetao Chen, Michael Milford, and Klaus McDonald-Maier, "A holistic visual place recognition approach using lightweight CNNs for significant viewpoint and appearance changes," *IEEE transactions on robotics*, vol. 36, no. 2, pp. 561–569, 2019. (pages 13 and 84).

[206] Sourav Garg, Niko Suenderhauf, and Michael Milford, "Lost? appearance-invariant place recognition for opposite viewpoints using visual semantics," in *proceedings of the robotics: science and systems (RSS)*, Pittsburgh, PA, USA, June 2018. (pages 13 and 84).

[207] Marvin Teichmann, Andre Araujo, Menglong Zhu, and Jack Sim, "Detect-to-retrieve: Efficient regional aggregation for image search," in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Long Beach, CA, USA, June 2019, pp. 5109–5118. (page 13).

[208] Mahesh Mohan, Dorian Gálvez-López, Claire Monteleoni, and Gabe Sibley, "Environment selection and hierarchical place recognition," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA, May 2015, pp. 5487–5494. (pages 14, 17, and 85).

[209] Loukas Bampis, Angelos Amanatiadis, and Antonios Gasteratos, "High order visual words for structure-aware and viewpoint-invariant loop closure detection," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Vancouver, Canada, September 2017, pp. 4268–4275. (page 14).

[210] Litao Yu, Adam Jacobson, and Michael Milford, "Rhythmic representations: Learning periodic patterns for scalable place recognition at a sublinear storage

cost," *IEEE robotics and automation letters*, vol. 3, no. 2, pp. 811–818, 2018. (pages 14, 86, and 87).

[211] Sourav Garg and Michael Milford, "Fast, compact and highly scalable visual place recognition through sequence-based matching of overloaded representations," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Paris, France, May 2020, pp. 3341–3348. (pages 14, 85, 86, and 87).

[212] Will Maddern, Michael Milford, and Gordon Wyeth, "CAT-SLAM: Probabilistic localisation and mapping using a continuous appearance-based trajectory," *The international journal of robotics research (SAGE)*, vol. 31, no. 4, pp. 429–451, 2012. (pages 14, 15, 17, and 86).

[213] Yang Liu and Hong Zhang, "Towards improving the efficiency of sequence-based SLAM," in *proceedings of the IEEE international conference on mechatronics and automation (ICMA)*, Takamatsu, Kagawa, Japan, August 2013, pp. 1261–1266. (pages 14 and 17).

[214] Niko Sünderhauf, Peer Neubert, and Peter Protzel, "Are we there yet? challenging seqslam on a 3000 km journey across all four seasons," in *proceedings of the IEEE international conference on robotics and automation workshop on long-term autonomy (ICRA)*, Karlsruhe, Germany, May 2013, p. 2013. (pages 14 and 27).

[215] Tayyab Naseer, Luciano Spinello, Wolfram Burgard, and Cyrill Stachniss, "Robust visual robot localization across seasons using network flows," in *proceedings of the AAAI conference on artificial intelligence*, Quebec City, QB, Canada, July 2014, pp. 2564–2570. (pages 14, 17, 81, and 82).

[216] Dongdong Bai, Chaoqun Wang, Bo Zhang, Xiaodong Yi, and Xuejun Yang, "Sequence searching with CNN features for robust and fast visual place recognition," *Computers and graphics (Elsevier)*, vol. 70, pp. 270–280, 2018. (page 14).

[217] A. Ranganathan and F. Dellaert, "Bayesian surprise and landmark detection," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Kobe, Japan, May 2009, pp. 2017–2023. (page 15).

[218] Christopher Mei, Gabe Sibley, and Paul Newman, "Closing loops without places," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Taipei, Taiwan, October 2010, pp. 3738–3744. (pages 15 and 16).

[219] Mikhail Volkov, Guy Rosman, Dan Feldman, John W Fisher, and Daniela Rus, "Coresets for visual summarization with applications to loop closure," in *proceed-*

ings of the IEEE international conference on robotics and automation (ICRA), Seattle, WA, USA, May 2015, pp. 3638–3645. (pages 15 and 86).

[220] Elena S Stumm, Christopher Mei, and Simon Lacroix, "Building location models for visual place recognition," *The international journal of robotics research (SAGE)*, vol. 35, no. 4, pp. 334–356, 2016. (pages 15 and 85).

[221] Edward Johns and Guang-Zhong Yang, "Feature co-occurrence maps: Appearance-based localisation throughout the day," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Karlsruhe, Germany, May 2013, pp. 3212–3218. (pages 15 and 83).

[222] Marvin Chancán, Luis Hernandez-Nunez, Ajay Narendra, Andrew B Barron, and Michael Milford, "A hybrid compact neural architecture for visual place recognition," *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 993–1000, 2020. (pages 15 and 82).

[223] Ananth Ranganathan, "PLISS: Labeling places using online changepoint detection," *Autonomous robots (Springer)*, vol. 32, no. 4, pp. 351–368, 2012. (page 15).

[224] Shan An, Guangfu Che, Fangru Zhou, Xianglong Liu, Xin Ma, and Yu Chen, "Fast and incremental loop closure detection using proximity graphs," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Macau, China, November 2019, pp. 378–385. (pages 15, 17, 27, and 87).

[225] Adrien Angeli, Stéphane Doncieux, Jean-Arcady Meyer, and David Filliat, "Real-time visual loop-closure detection," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Pasadena, CA, USA, May 2008, pp. 1842–1847. (page 15).

[226] Herve Jegou, Matthijs Douze, and Cordelia Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *proceedings of the European conference on computer vision (ECCV)*, Marseille, France, October 2008, pp. 304–317. (pages 15 and 16).

[227] Henrik Stewénius, Steinar H Gunderson, and Julien Pilet, "Size matters: exhaustive geometric verification for image retrieval," in *proceedings of the European conference on computer vision (ECCV)*, Florence, Italy, October 2012, pp. 674–687. (page 15).

[228] Titus Cieslewski, Elena Stumm, Abel Gawel, Mike Bosse, Simon Lynen, and Roland Siegwart, "Point cloud descriptors for place recognition using sparse visual information," in *proceedings of the IEEE international conference on*

robotics and automation (ICRA), Stockholm, Sweden, May 2016, pp. 4830–4836. (pages 15, 16, and 85).

[229] Brian Williams, Mark Cummins, José Neira, Paul Newman, Ian Reid, and Juan Tardós, "A comparison of loop closing techniques in monocular SLAM," *Robotics and autonomous systems (Elsevier)*, vol. 57, no. 12, pp. 1188–1197, 2009. (page 15).

[230] Kin Leong Ho and Paul Newman, "Detecting loop closure with scene sequences," *International journal of computer vision (Springer)*, vol. 74, no. 3, pp. 261–286, 2007. (page 16).

[231] Ben JA Kröse, Nikos Vlassis, Roland Bunschoten, and Yoichi Motomura, "A probabilistic model for appearance-based robot localization," *Image and vision computing (Elsevier)*, vol. 19, no. 6, pp. 381–391, 2001. (page 17).

[232] Peter Hansen and Brett Browning, "Visual place recognition using HMM sequence matching," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Chicago, IL USA, September 2014, pp. 4549–4555. (pages 17 and 81).

[233] Roberto Arroyo, Pablo F Alcantarilla, Luis M Bergasa, and Eduardo Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Seattle, WA, USA, May 2015, pp. 6328–6335. (pages 17 and 81).

[234] Peer Neubert, Stefan Schubert, and Peter Protzel, "A neurologically inspired sequence processing model for mobile robot place recognition," *IEEE robotics and automation letters*, vol. 4, no. 4, pp. 3200–3207, 2019. (page 17).

[235] Lawrence Rabiner, "Fundamentals of speech recognition," in *Fundamentals of speech recognition*. PTR Prentice Hall, 1993. (page 17).

[236] Ben Talbot, Sourav Garg, and Michael Milford, "OpenSeqSLAM2.0: An open source toolbox for visual place recognition under changing conditions," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Madrid, Spain, October 2018, pp. 7758–7765. (pages 17, 27, and 46).

[237] Andrew Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE transactions on information theory*, vol. 13, no. 2, pp. 260–269, 1967. (page 17).

[238] Navid Nourani-Vatani and Cedric Pradalier, “Scene change detection for vision-based topological mapping and localization,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Taipei, Taiwan, October 2010, pp. 3792–3797. (page 17).

[239] Hao Zhang, Fei Han, and Hua Wang, “Robust multimodal sequence-based loop closure detection via structured sparsity.” in *proceedings of the robotics: science and systems (RSS)*, Ann Arbor, MI, USA, June 2016. (page 17).

[240] Andrea Banino, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J Chadwick, Thomas Degris, Joseph Modayil, *et al.*, “Vector-based navigation using grid-like representations in artificial agents,” *Nature*, vol. 557, no. 7705, pp. 429–433, 2018. (page 17).

[241] Xiwu Zhang, Lei Wang, Yan Zhao, and Yan Su, “Graph-based place recognition in image sequences with CNN features,” *Journal of intelligent and robotic systems (Springer)*, vol. 95, no. 2, pp. 389–403, 2019. (pages 17 and 83).

[242] Martin A Fischler and Robert C Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981. (page 18).

[243] David Nistér, “An efficient solution to the five-point relative pose problem,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 6, pp. 756–770, 2004. (page 18).

[244] Brian Williams, Mark Cummins, José Neira, Paul Newman, Ian Reid, and Juan Tardós, “An image-to-map loop closing method for monocular SLAM,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nice, France, September 2008, pp. 2053–2059. (page 18).

[245] Zaïd Harchaoui and Francis Bach, “Image classification with segmentation graph kernels,” in *proceedings of the computer society conference on computer vision and pattern recognition (CVPR)*, Minneapolis, MI, USA, June 2007, pp. 1–8. (page 18).

[246] Kurt Konolige, James Bowman, JD Chen, Patrick Mihelich, Michael Calonder, Vincent Lepetit, and Pascal Fua, “View-based maps,” *The international journal of robotics research (SAGE)*, vol. 29, no. 8, pp. 941–957, 2010. (page 18).

[247] Paul J Besl and Neil D McKay, “Method for registration of 3-D shapes,” in *proceedings of the sensor fusion IV: control paradigms and data structures*, Boston, MA, USA, April 1992, pp. 586–606. (page 18).

[248] David MW Powers, “Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation,” *arXiv preprint arXiv:2010.16061*, 2020. (page 21).

[249] Jesse Davis and Mark Goadrich, “The relationship between Precision-Recall and ROC curves,” in *proceedings of the international conference on machine learning*, Pittsburgh, PA, USA, June 2006, pp. 233–240. (page 22).

[250] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun, “Vision meets robotics: The KITTI dataset,” *The international journal of robotics research (SAGE)*, vol. 32, no. 11, pp. 1231–1237, 2013. (pages 23, 24, 35, 36, 47, 49, 50, 51, 68, 69, 73, 77, and 79).

[251] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart, “The EuRoC micro aerial vehicle datasets,” *The international journal of robotics research (SAGE)*, vol. 35, no. 10, pp. 1157–1163, 2016. (pages 25, 77, and 79).

[252] Jose-Luis Blanco, Francisco-Angel Moreno, and Javier Gonzalez, “A collection of outdoor robotic datasets with centimeter-accuracy ground truth,” *Autonomous robots (Springer)*, vol. 27, no. 4, pp. 327–351, 2009. (pages 25, 47, 49, 77, and 79).

[253] Shan An, Haogang Zhu, Dong Wei, Konstantinos A Tsintotas, and Antonios Gasteratos, “Fast and incremental loop closure detection with deep features and proximity graphs,” *arXiv preprint arXiv:2010.11703*, 2020. (pages 25, 27, and 87).

[254] Mike Smith, Ian Baldwin, Winston Churchill, Rohan Paul, and Paul Newman, “The New College vision and laser data set,” *The international journal of robotics research (SAGE)*, vol. 28, no. 5, pp. 595–599, 2009. (pages 26, 47, 49, 50, and 77).

[255] Loukas Bampis, Angelos Amanatiadis, and Antonios Gasteratos, “Fast loop-closure detection using visual-word-vectors from image sequences,” *The international journal of robotics research (SAGE)*, vol. 37, no. 1, pp. 62–82, 2018. (page 27).

[256] Bernd Fritzke *et al.*, “A growing neural gas network learns topologies,” in *proceedings of the advances in neural information processing systems (NIPS)*, Denver, CO, USA, November 1995, pp. 625–632. (pages 31 and 32).

[257] Bruce D Lucas, Takeo Kanade, *et al.*, “An iterative image registration technique with an application to stereo vision,” in *proceedings of the international joint*

conference on artificial intelligence, Vancouver, BC, Canada, August 1981, pp. 674–679. (pages 39, 40, 43, and 57).

[258] Thorsten Thormahlen, Nils Hasler, Michael Wand, and H-P Seidel, "Merging of feature tracks for camera motion estimation from video," in *proceedings of the IET European conference on visual media production (CVMP)*, London, UK, November 2008. (page 41).

[259] Geert De Cubber and Hichem Sahli, "Partial differential equation-based dense 3d structure and motion estimation from monocular image sequences," *IET computer vision*, vol. 6, no. 3, pp. 174–185, 2012. (page 41).

[260] Roziana Ramli, Mohd Yamani Idna Idris, Khairunnisa Hasikin, Noor Khairiah A Karim, Ainuddin Wahid Abdul Wahab, Ismail Ahmedy, Fatimah Ahmedy, and Hamzah Arof, "Local descriptor for retinal fundus image registration," *IET Computer Vision*, vol. 14, no. 4, pp. 144–153, 2020. (page 41).

[261] Raul Mur-Artal and Juan D Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017. (page 57).

[262] Jon Louis Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975. (pages 59 and 87).

[263] Yang Liu and Hong Zhang, "Indexing visual features: Real-time loop closure detection using a tree structure," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Saint Paul, MN, USA, May 2012, pp. 3613–3618. (pages 59, 86, and 87).

[264] Artem Babenko and Victor Lempitsky, "The inverted multi-index," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 6, pp. 1247–1260, 2014. (pages 59 and 87).

[265] Dominik Schlegel and Giorgio Grisetti, "HBST: A hamming distance embedding binary search tree for feature-based visual place recognition," *IEEE robotics and automation letters*, vol. 3, no. 4, pp. 3741–3748, 2018. (pages 59 and 86).

[266] Ana C Murillo, Gautam Singh, Jana Kosecká, and José Jesús Guerrero, "Localization in urban environments using a panoramic gist descriptor," *IEEE transactions on robotics*, vol. 29, no. 1, pp. 146–160, 2012. (page 61).

[267] Richard Durbin, Sean R Eddy, Anders Krogh, and Graeme Mitchison, *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge university press, 1998. (page 67).

[268] Will Maddern, Geoffrey Pascoe, Chris Linegar, and Paul Newman, “1 year, 1000 km: The Oxford RobotCar dataset,” *The international journal of robotics research (SAGE)*, vol. 36, no. 1, pp. 3–15, 2017. (page 82).

[269] Eli Shechtman and Michal Irani, “Matching local self-similarities across images and videos,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Minneapolis, MN, USA, June 2007, pp. 1–8. (page 81).

[270] Vassilios Vonikakis, Rigas Kouskouridas, and Antonios Gasteratos, “On the evaluation of illumination compensation algorithms,” *Multimedia tools and applications*, vol. 77, no. 8, pp. 9211–9231, 2018. (page 81).

[271] Winston Churchill and Paul Newman, “Practice makes perfect? managing and leveraging visual experiences for lifelong navigation,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Saint Paul, MN, USA, May 2012, pp. 4525–4532. (page 82).

[272] Roberto Arroyo, Pablo F Alcantarilla, Luis M Bergasa, and Eduardo Romera, “Fusion and binarization of CNN features for robust topological localization across seasons,” in *proceedings of the IEEE international conference on intelligent robots and systems (IROS)*, Daejeon, Korea (South), October 2016, pp. 4656–4663. (pages 82 and 88).

[273] Christoffer Valgren and Achim J Lilienthal, “SIFT, SURF & seasons: Appearance-based long-term localization in outdoor environments,” *Robotics and autonomous systems (Elsevier)*, vol. 58, no. 2, pp. 149–156, 2010. (page 82).

[274] Colin McManus, Ben Upcroft, and Paul Newmann, “Scene signatures: Localised and point-less features for localisation,” in *proceedings of the robotics: science and systems (RSS)*, Rome, Italy, July 2014, pp. 1–9. (pages 82 and 83).

[275] Peer Neubert, Niko Sünderhauf, and Peter Protzel, “Appearance change prediction for long-term navigation across seasons,” in *proceedings of the European conference on mobile robots*, Barcelona, Spain, September 2013, pp. 198–203. (pages 82 and 83).

[276] Stephanie M Lowry, Michael J Milford, and Gordon F Wyeth, “Transforming morning to afternoon using linear regression techniques,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, June 2014, pp. 3950–3955. (pages 82 and 83).

[277] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva, “Learning deep features for scene recognition using places database,” in

proceedings of the advances in neural information processing systems (NIPS), Montreal,QB, Canada, December 2014. (page 82).

[278] Winston Churchill and Paul Newman, “Experience-based navigation for long-term localisation,” *The international journal of robotics research (SAGE)*, vol. 32, no. 14, pp. 1645–1661, 2013. (pages 82 and 83).

[279] Stephanie M Lowry, Gordon F Wyeth, and Michael J Milford, “Towards training-free appearance-based localization: probabilistic models for whole-image descriptors,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, June 2014, pp. 711–717. (page 82).

[280] Zetao Chen, Stephanie Lowry, Adam Jacobson, Zongyuan Ge, and Michael Milford, “Distance metric learning for feature-agnostic place recognition,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Hamburg, Germany, September 2015, pp. 2556–2563. (page 82).

[281] S Vidas and W Maddern, “Towards robust night and day place recognition using visible and thermal imaging,” in *proceedings of the robotics: science and systems (RSS)*, Sydney, NSW, Australia, July 2012. (page 82).

[282] Assia Benbihi, Stéphanie Arravechia, Matthieu Geist, and Cédric Pradalier, “Image-based place recognition on bucolic environment across seasons from semantic edge description,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Paris, France, May 2020, pp. 3032–3038. (page 82).

[283] Xuming He, Richard S Zemel, and Volodymyr Mnih, “Topological map learning from outdoor image sequences,” *Journal of field robotics*, vol. 23, no. 11-12, pp. 1091–1104, 2006. (page 83).

[284] Chris Linegar, Winston Churchill, and Paul Newman, “Made to measure: Bespoke landmarks for 24-hour, all-weather localisation with a camera,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Stockholm, Sweden, May 2016, pp. 787–794. (page 83).

[285] Peer Neubert, Niko Sünderhauf, and Peter Protzel, “Superpixel-based appearance change prediction for long-term navigation across seasons,” *Robotics and autonomous systems*, vol. 69, pp. 15–27, 2015. (page 83).

[286] Stephanie Lowry and Michael J Milford, “Supervised and unsupervised linear learning techniques for visual place recognition in changing environments,” *IEEE transactions on robotics*, vol. 32, no. 3, pp. 600–613, 2016. (page 83).

[287] José M Álvarez Alvarez and Antonio M Ĺopez, “Road detection based on illuminant invariance,” *IEEE transaction on intelligent transportation systems*, vol. 12, no. 1, pp. 184–193, 2010. (page 83).

[288] Ananth Ranganathan, Shohei Matsumoto, and David Ilstrup, “Towards illumination invariance for visual localization,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Karlsruhe, Germany, May 2013, pp. 3791–3798. (page 83).

[289] Peter Corke, Rohan Paul, Winston Churchill, and Paul Newman, “Dealing with shadows: Capturing intrinsic scene appearance for image-based outdoor localisation,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Tokyo, Japan, November 2013, pp. 2085–2092. (page 83).

[290] Moein Shakeri and Hong Zhang, “Illumination invariant representation of natural images for visual place recognition,” in *proceedings of the IEEE international conference on intelligent robots and systems (IROS)*, Daejeon, Korea (South), October 2016, pp. 466–472. (page 83).

[291] Henning Lategahn, Johannes Beck, Bernd Kitt, and Christoph Stiller, “How to learn an illumination robust image feature for place recognition,” in *proceedings of the IEEE intelligent vehicles symposium (IV)*, Gold Coast, QLD, Australia, June 2013, pp. 285–291. (page 83).

[292] Mubariz Zaffar, Shoaib Ehsan, Michael Milford, and Klaus McDonald-Maier, “CoHOG: A light-weight, compute-efficient, and training-free visual place recognition technique for changing environments,” *IEEE robotics and automation letters*, vol. 5, no. 2, pp. 1835–1842, 2020. (page 83).

[293] Horia Porav, Will Maddern, and Paul Newman, “Adversarial training for adverse conditions: Robust metric localisation using appearance transfer,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Brisbane, Australia, May 2018, pp. 1011–1018. (page 83).

[294] Michael Milford, Chunhua Shen, Stephanie Lowry, Niko Suenderhauf, Sareh Shirazi, Guosheng Lin, Fayao Liu, Edward Pepperell, Cesar Lerma, Ben Upcroft, *et al.*, “Sequence searching with deep-learnt depth for condition-and viewpoint-invariant route-based place recognition,” in *proceedings of the IEEE conference on computer vision and pattern recognition workshop (CVPR)*, Boston, MA, USA, June 2015, pp. 18–25. (page 83).

[295] Andrzej Pronobis, Barbara Caputo, Patric Jensfelt, and Henrik I Christensen, “A discriminative approach to robust visual place recognition,” in *proceedings of*

the IEEE/RSJ international conference on intelligent robots and systems (IROS), Beijing, China, October 2006, pp. 3829–3836. (page 83).

[296] Roberto Arroyo, Pablo F Alcantarilla, Luis M Bergasa, J Javier Yebes, and Sergio Gámez, “Bidirectional loop closure detection on panoramas for visual navigation,” in *proceedings of the IEEE intelligent vehicles symposium*, Dearborn, MI, USA, June 2014, pp. 1378–1383. (page 84).

[297] Xiangru Li and Zhanyi Hu, “Rejecting mismatches by correspondence function,” *International journal of computer vision (Springer)*, vol. 89, no. 1, pp. 1–17, 2010. (page 84).

[298] Sourav Garg, Niko Suenderhauf, and Michael Milford, “Semantic–geometric visual place recognition: a new perspective for reconciling opposing views,” *The international journal of robotics research (SAGE)*, p. 0278364919839761, 2019. (page 84).

[299] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid, “Refinenet: Multi-path refinement networks for high-resolution semantic segmentation,” in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Honolulu, HI, USA, July 2017, pp. 1925–1934. (page 84).

[300] Georg Klein and David Murray, “Parallel tracking and mapping for small AR workspaces,” in *proceedings of the IEEE and ACM international symposium on mixed and augmented reality*, Nara, Japan, November 2007, pp. 225–234. (page 84).

[301] Adrien Angeli, Stéphane Doncieux, Jean-Arcady Meyer, and David Filliat, “Incremental vision-based topological SLAM,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nice, France, September 2008, pp. 1031–1036. (page 85).

[302] Hordur Johannsson, Michael Kaess, Maurice Fallon, and John J Leonard, “Temporally scalable visual SLAM using a reduced pose graph,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Karlsruhe, Germany, May 2013, pp. 54–61. (page 85).

[303] Raúl Mur-Artal and Juan D Tardós, “Fast relocalisation and loop closing in keyframe-based slam,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, June 2014, pp. 846–853. (page 85).

[304] Hong Zhang, Bo Li, and Dan Yang, “Keyframe detection for appearance-based visual SLAM,” in *proceedings of the IEEE/RSJ international conference on*

intelligent robots and systems (IROS), Taipei, Taiwan, October 2010, pp. 2071–2076. (page 85).

[305] H. Strasdat, A. J. Davison, J. M. M. Montiel, and K. Konolige, "Double window optimisation for constant time visual SLAM," in *proceedings of the IEEE international conference on computer vision*, Barcelona, Spain, November 2011, pp. 2352–2359. (pages 85 and 87).

[306] Mubariz Zaffar, Shoaib Ehsan, Michael Milford, and Klaus D McDonald-Maier, "Memorable maps: A framework for re-defining places in visual place recognition," *IEEE transactions on intelligent transportation systems*, pp. 1–15, 2020. (page 85).

[307] Carlos Estrada, José Neira, and Juan D Tardós, "Hierarchical SLAM: Real-time accurate mapping of large environments," *IEEE transactions on robotics*, vol. 21, no. 4, pp. 588–596, 2005. (page 85).

[308] Zoran Zivkovic, Bram Bakker, and Ben Krose, "Hierarchical map building using visual landmarks and geometric constraints," in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Edmonton, AB, Canada, August 2005, pp. 2480–2485. (page 85).

[309] Olaf Booij, Bas Terwijn, Zoran Zivkovic, and B Krose, "Navigation using an appearance based topological map," in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Rome, Italy, April 2007, pp. 3927–3932. (page 85).

[310] Olaf Booij, Zoran Zivkovic, and Ben Kröse, "Efficient data association for view based SLAM using connected dominating sets," *Robotics and autonomous systems*, vol. 57, no. 12, pp. 1225–1234, 2009. (page 85).

[311] Giorgio Grisetti, Rainer Kümmerle, Cyrill Stachniss, Udo Frese, and Christoph Hertzberg, "Hierarchical optimization on manifolds for online 2D and 3D mapping," in *proceedings of the IEEE international conference on robotics and automation workshop (ICRA)*, Anchorage, Alaska, May 2010, pp. 273–278. (page 85).

[312] Zetao Chen, Stephanie Lowry, Adam Jacobson, Michael E Hasselmo, and Michael Milford, "Bio-inspired homogeneous multi-scale place recognition," *Neural networks*, vol. 72, pp. 48–61, 2015. (page 85).

[313] Xiaohan Fei, Konstantine Tsotsos, and Stefano Soatto, "A simple hierarchical pooling data structure for loop closure," in *proceedings of the European conference on computer vision (ECCV)*, Amsterdam, the Netherlands, October 2016, pp. 321–337. (page 85).

[314] Stephen Hausler and Michael Milford, “Hierarchical multi-process fusion for visual place recognition,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Paris, France, May 2020, pp. 3327–3333. (page 85).

[315] Hanne Stensola, Tor Stensola, Trygve Solstad, Kristian Frøland, May-Britt Moser, and Edvard I Moser, “The entorhinal grid map is discretized,” *Nature*, vol. 492, no. 7427, pp. 72–78, 2012. (page 85).

[316] Norbert Kruger, Peter Janssen, Sinan Kalkan, Markus Lappe, Ales Leonardis, Justus Piater, Antonio J Rodriguez-Sanchez, and Laurenz Wiskott, “Deep hierarchies in the primate visual cortex: What can we learn for computer vision?” *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1847–1871, 2012. (page 85).

[317] Naftali Tishby, Fernando C Pereira, and William Bialek, “The information bottleneck method,” *arXiv preprint physics/0004057*, 2000. (pages 85 and 87).

[318] Liz Murphy and Gabe Sibley, “Incremental unsupervised topological place discovery,” in *proceedings of the IEEE international conference on robotics and automation (ICRA)*, Hong Kong, China, June 2014, pp. 1312–1318. (page 85).

[319] Michael Milford and Gordon Wyeth, “Persistent navigation and mapping using a biologically inspired slam system,” *The international journal of robotics research (SAGE)*, vol. 29, no. 9, pp. 1131–1153, 2010. (page 85).

[320] Oguzhan Guclu and Ahmet Burak Can, “Fast and effective loop closure detection to improve SLAM performance,” *Journal of intelligent and robotic systems (Springer)*, vol. 93, no. 3-4, pp. 495–517, 2019. (page 85).

[321] Feras Dayoub and Tom Duckett, “An adaptive appearance-based map for long-term topological localization of mobile robots,” in *proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Nice, France, September 2008, pp. 3364–3369. (page 86).

[322] Mathieu Labbe and François Michaud, “Online global loop closure detection for large-scale multi-session graph-based SLAM,” in *proceedings of the 2014 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Chicago, IL, USA, September 2014, pp. 2661–2666. (page 86).

[323] Stephanie Lowry and Henrik Andreasson, “Lightweight, viewpoint-invariant visual place recognition in changing environments,” *IEEE Robotics and automation letters*, vol. 3, no. 2, pp. 957–964, 2018. (page 86).

[324] Yi Hou, Hong Zhang, and Shilin Zhou, “Tree-based indexing for real-time convnet landmark-based visual place recognition,” *International journal of advanced robotic systems (SAGE)*, vol. 14, no. 1, p. 1729881416686951, 2017. (page 86).

[325] Jing Wang, Jingdong Wang, Gang Zeng, Zhuowen Tu, Rui Gan, and Shipeng Li, “Scalable k-nn graph construction for visual descriptors,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Providence, RI, USA, June 2012, pp. 1106–1113. (page 86).

[326] Ben Harwood and Tom Drummond, “FANNG: Fast approximate nearest neighbour graphs,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Las Vegas, NV, USA, June 2016, pp. 5713–5722. (page 86).

[327] Herve Jegou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Perez, and Cordelia Schmid, “Aggregating local image descriptors into compact codes,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 9, pp. 1704–1716, 2011. (page 86).

[328] Jingdong Wang, Ting Zhang, Nicu Sebe, Heng Tao Shen, *et al.*, “A survey on learning to hash,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 769–790, 2017. (page 87).

[329] Aristides Gionis, Piotr Indyk, Rajeev Motwani, *et al.*, “Similarity search in high dimensions via hashing,” in *proceedings of the international conference on very large data bases (VLDB)*, Edinburgh, Scotland, UK, September 1999, pp. 518–529. (page 87).

[330] Ruslan Salakhutdinov and Geoffrey Hinton, “Semantic hashing,” *International journal of approximate reasoning (Elsevier)*, vol. 50, no. 7, pp. 969–978, 2009. (page 87).

[331] Jiayi Ma, Ji Zhao, Junjun Jiang, Huabing Zhou, and Xiaojie Guo, “Locality preserving matching,” *International journal of computer vision (Springer)*, vol. 127, no. 5, pp. 512–531, 2019. (page 87).

[332] Xingyu Jiang, Jiayi Ma, Junjun Jiang, and Xiaojie Guo, “Robust feature matching using spatial clustering with heavy outliers,” *IEEE transactions on image processing*, vol. 29, pp. 736–746, 2019. (page 87).

[333] Deepak Ravichandran, Patrick Pantel, and Eduard Hovy, “Randomized algorithms and NLP: Using locality sensitive hash functions for high speed noun clustering,” in *proceedings of the annual meeting on association for computational linguistics*, Ann Arbor, MI, USA, June 2005, pp. 622–629. (page 87).

[334] Marius Muja and David G Lowe, “Fast matching of binary features,” in *proceedings of the IEEE conference on computer and robot vision*, Toronto, ON, Canada, May 2012, pp. 404–410. (page 87).

[335] David Nister and Henrik Stewenius, “Scalable recognition with a vocabulary tree,” in *proceedings of the computer society conference on computer vision and pattern recognition (CVPR)*, New York, NY, USA, June 2006, pp. 2161–2168. (page 87).

[336] Hanan Samet, *The design and analysis of spatial data structures*. Addison-Wesley, Reading, MA, USA, 1990, vol. 85. (page 87).

[337] Marius Muja and David G Lowe, “Fast approximate nearest neighbors with automatic algorithm configuration,” in *proceedings of the international conference on computer vision theory and applications (VISAPP)*, Lisboa, Portugal, February 2009, pp. 331–340. (page 87).

[338] C Chow and Cong Liu, “Approximating discrete probability distributions with dependence trees,” *IEEE transactions on information theory*, vol. 14, no. 3, pp. 462–467, 1968. (page 87).

[339] Donald Geman and Bruno Jedynak, “An active testing model for tracking roads in satellite images,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 18, no. 1, pp. 1–14, 1996. (page 87).

[340] Kristen Grauman and Trevor Darrell, “The pyramid match kernel: Discriminative classification with sets of image features,” in *proceedings of the IEEE international conference on computer vision (ICCV)*, Beijing, China, October 2005, pp. 1458–1465. (page 87).

[341] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *proceedings of the computer society conference on computer vision and pattern recognition (CVPR)*, vol. 2, New York, NY, USA, June 2006, pp. 2169–2178. (page 87).

[342] Chanop Silpa-Anan and Richard Hartley, “Optimised kd-trees for fast image descriptor matching,” in *proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, Anchorage, AL, USA, June 2008, pp. 1–8. (page 87).

[343] Aram Kawewong, Noppharit Tongprasit, and Osamu Hasegawa, “PIRF-Nav 2.0: Fast and online incremental appearance-based loop-closure detection in an indoor environment,” *Robotics and autonomous systems (Elsevier)*, vol. 59, no. 10, pp. 727–739, 2011. (page 87).

[344] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba, “Places: A 10 million image database for scene recognition,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1452–1464, 2017. (page 88).

Thesis Publications

The core of this Ph.D. dissertation is supported by the following publications.

Journals:

1. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Probabilistic appearance-based place recognition through bag of tracked words," *IEEE robotics and automation letters*, vol. 4, no. 2, pp. 1737–1744, 2019. *Presented at the IEEE international conference on robotics and automation (ICRA)*, May, 2019, Montreal (Canada).

2. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Tracking-DOSeqSLAM: A dynamic sequence-based visual place recognition paradigm," *IET computer vision (Wiley)*, vol. 15, no. 4, pp. 258-273, 2021.

3. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Modest-vocabulary loop-closure detection with incremental bag of tracked words," *Robotics and autonomous systems (Elsevier)*, vol. 141, p. 103782, 2021.

4. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “The revisiting problem in simultaneous localization and mapping: A survey on visual loop closure detection," *Under revision with the* *IEEE transactions on intelligent transportation systems*, 2021.

Conferences:

1. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Assigning visual words to places for loop closure detection," *in proceedings of the IEEE international conference on robotics and automation (ICRA)*, Brisbane, Australia, May 2018, pp. 5979–5985.

2. **Konstantinos A. Tsintotas**, Loukas Bampis, Stelios Rallis, and Antonios Gasteratos, “SeqSLAM with bag of visual words for appearance based loop closure detection," *in proceedings of the international conference on robotics in Alpe-Adria Danube region (RAAD)*, Patras, Greece, June 2018, pp. 580–587.

3. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “DOSeqSLAM: dynamic on-line sequence based loop closure detection algorithm for SLAM," *in proceedings of the IEEE international conference on imaging systems and techniques (IST)*, Krakow, Poland, October 2018, pp. 1–6.

4. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Appearance-based loop closure detection with scale-restrictive visual features," *in proceedings of the international conference on computer vision systems (ICVS)*, Thessaloniki, Greece, September 2019, pp. 75–87.

5. **Konstantinos A. Tsintotas**, Loukas Bampis, Shan An, George F. Fragulis, Spyridon G. Mouroutsos, and Antonios Gasteratos, “Sequence-based mapping for probabilistic visual loop closure detection," *Accepted for publication* *in proceedings of the IEEE international conference on imaging systems and techniques (IST)*, New York, USA, August 2021.

Book chapters:

1. **Konstantinos A. Tsintotas**, Loukas Bampis, and Antonios Gasteratos, “Visual place recognition for simultaneous localization and mapping," *submitted to* George Giakos (Editor), *Multifaceted imaging principles and augmented intelligence. Springer.*

Other Publications

The following publications arose during the author's PhD studies and contributed to the conception of the presented approaches.

Journals:

1. Shan An, Haogang Zhu, Dong Wei, **Konstantinos A. Tsintotas**, and Antonios Gasteratos, "Fast and incremental loop closure detection with deep features and proximity graphs,", *Submitted to Journal of field robotics (Wiley)*, 2021.

Conferences:

1. Shan An, Fangru Zhou, Mei Yang, Haogang Zhu, Changhong Fu, and **Konstantinos A. Tsintotas**, "Real-time monocular pedestrian depth estimation and segmentation on embedded systems," *Accepted for publication in proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Prague, Czech Republic, September 2021.
2. **Konstantinos A. Tsintotas**, Loukas Bampis, Anastasios Taitzoglou, Ioannis Kansizoglou, and Antonios Gasteratos, "Safe UAV landing: A low-complexity pipeline for surface conditions recognition." *Accepted for publication in proceedings of the IEEE international conference on imaging systems and techniques (IST)*, New York, USA, August 2021. Awarded as best student paper.
3. Shan An, Xiajie Zhang, Dong Wei, Haogang Zhu, Jianyu Yang, and **Konstantinos A. Tsintotas**, "FastHand: Fast hand pose estimation from a monocular camera," *Submitted to the IEEE international conference on embedded software and systems (ICESS)*, 2021.