Detection-by-Localization: Maintenance-Free Change Object Detector

Tanaka Kanji

Abstract

Recent researches demonstrate that self-localization performance is a very useful measure of likelihood-of-change (LoC) for change detection. In this paper, this “detection-by-localization” scheme is studied in a novel generalized task of object-level change detection. In our framework, a given query image is segmented into object-level subimages (termed “scene parts”), which are then converted to subimage-level pixel-wise LoC maps via the detection-by-localization scheme. Our approach models a self-localization system as a ranking function, outputting a ranked list of reference images, without requiring relevance score. Thanks to this new setting, we can generalize our approach to a broad class of self-localization systems. Our ranking based self-localization model allows to fuse self-localization results from different modalities via an unsupervised rank fusion derived from a field of multi-modal information retrieval (MMR). This study is a first step towards a maintenance-free approach, minimizing maintenance cost of map maintenance system (e.g., background model, detector engine).

I. INTRODUCTION

To maintain an environment map in dynamic environments, a robotic visual SLAM system must be capable of detecting changed objects on the map. To be effective, a map maintenance capability must fulfill two requirements: the robot must know where it is (i.e., map-relative robot self-localization), and it must differentiate between changed objects and nuisances (i.e., robot-centric change detection). Motivated by the recent maturity of self-localization techniques (e.g., visual place recognition, loop-closure detection, map-matching), we propose the reuse of self-localization capability to address the latter requirement (i.e., change detection).

Recent studies demonstrated that self-localization performance is a very useful measure of the likelihood of change (LoC) for change detection. The experience-based mapping framework originally proposed in [1] is a good example. In that framework, an environment map was permanently augmented by multiple sub-maps (i.e., visual experiences) using data from differing environment conditions. If self-localization with such a map performs sufficiently well, it covers the encountered conditions and there is no need to update the map (i.e., low LoC). However, if self-localization performance is poor, the map does not cover the encountered conditions during the sortie (i.e., high LoC).

Such a new change detection scheme, referred to as “detection-by-localization” in the current study, has two main advantages:

1) No additional storage or detector engine (but only existing map database and localization engine) is required;
2) The degradation of map quality (i.e., need of map update) in terms of self-localization performance can be directly detected.

This paper addresses a new detection-by-localization task that detects changes at subimage- or object- levels (Fig. 1). Unlike previous methods that classify a given query image as “change” or “no-change”, we not only classify a query image but also detect changed objects (in the form of bounding boxes) within the image frame. The relationship between image- and object-level change detection is analogous to the relationship between image [2] and object localization [3] in the computer vision literature. Object-level detection becomes an even more challenging task than image-level detection, owning to the requirements of object segmentation and low image resolution. Importantly, object-level change detection is common in robotics and has many important applications, including patrol robot in partially changing environments [4] and object segmentation via change detection [5]. However, object-level change detection has not been explored in the context of detection-by-localization.

In our task scenario, a ranked list of reference or background images retrieved from a map database by a self-localization system is viewed as the measure of self-localization performance. Formally, we consider a detection-by-localization scenario in which reliable viewpoint measurement (referred to as “ground-truth” viewpoint) is available [1], and we measure self-localization performance (i.e., LoC) by the rank value assigned to the “ground-truth” reference image. The self-localization model is inspired by our previous work on unsupervised part-based scene modeling, in which the popular bag-of-visual-features (BoVF) self-localization scheme [2] is extended in two ways.

1) Instead of point features (e.g., SIFT), discriminative subimage-level features (referred to as “scene parts”) are used to model a query/reference scene.
2) Reference subimages are retrieved using query subimages, and are aggregated into an image-level decision (i.e., ranked list).

In our contribution,

Our work has been supported in part by JSPS KAKENHI Grant-in-Aid for Scientific Research (C) 26330297, and (C) 17K00361.
K. Tanaka is with Faculty of Engineering, University of Fukui, Japan. tkknj@u-fukui.ac.jp
We would like to express our sincere gratitude to Takuma Sugimoto, Rino Ide and Kousuke Yamaguchi for development of deep learning architecture, and initial investigation on change detection tasks on the dataset, which helped us to focus on our Detection-by-Localization project.
In the detection-by-localization scheme, self-localization performance (i.e., rank value of the “ground-truth” reference image in the map database) is used as a measure of likelihood-of-change (LoC) for change detection. The new generalized problem of object-level change detection, addressed in this paper, takes object-level subimages (instead of a full image) as query input for the self-localization, and outputs a subimage-level pixel-wise LoC map.

- we explain how such an image-level self-localization model can be adopted to the novel task of subimage-level change detection,
- by addressing how a ranked list can be interpreted as a subimage-level pixel-wise LoC map,
- and by exploring ways of aggregating subimage-level LoC maps into the final decision of a single image-level LoC map.

Our unsupervised ranking based approach is an advantage, because it does not require training examples on the image collection. This is very important, because we must deal with increasingly complex maps and images, and automatic collection of their ground-truth data by the robot-self is not a trivial task.

Presently, most image change detection algorithms [6] focus on pairwise image comparison, to differentiate a given query-background-image-pair. It is straightforward to adopt such a method to object-level change detection by introducing an image segmentation preprocessing step [7]. This has recently achieved state-of-the-art performance with a weakly-supervised setting [8]. However, these image-to-image differencing methods are not directly applicable to real-time SLAM systems. First, they are too expensive (on average) to compute in real-time. Moreover, they require memorization and maintenance of many background images proportional to the map size. Some alternative popular approaches are those that classify a sole query image instead of a query-background-image-pair as a change or a no-change, compared to an offline pretrained background model. Such an approach has the potential to realize model compactness by using compressed background models such as BoVFs, geometric model and compact manifold learning. However, its storage overhead is proportional to map size and increases in unbounded fashion. Moreover, updating this background model significantly complicates the map updating process. To suppress such computational time and space complexities by reusing existing resources of map database and localization engine is an objective of the current study.

II. DETECTION-BY-LOCALIZATION FRAMEWORK

We consider a new detection-by-localization framework consisting of a self-localization system and a change detection system, in which self-localization performance (i.e., rank value of the “ground-truth” reference image as explained in Section I) is used as an LoC measure for change detection. Both self-localization and change detection can be confused by differing fields-of-view (FoV), occlusions, and seasonal changes of object appearance or appearing of visible objects.

A. Scene Model

The scene model is assumed to be a BoVF scene model [2]. The vast majority of real-time visual SLAM systems (i.e., iBOW-LCD [9]) employs BoVF models. Whereas other scene modeling schemes such as the deep ConvNet classifier have been studied in the self-localization literature. The BoVF scheme is advantageous to computational speed, compactness, and discriminativity, and has been a de facto standard method for real-time self-localization tasks, such as loop-closure detection.
Fig. 2. Estimating LoC map. We consider the new setting where each pixel may belong to multiple subimages (referred to as qBBs). Each subimage is then input to a self-localization system to obtain a rank value, and then the rank value of each pixel is computed from these subimages which the pixel belongs to. In the figure, a rectangle indicates the bounding box of an image or a subimage, and a ‘+’ mark indicates the pixel of interest.

B. Self-Localization Model

The self-localization system is modeled as a ranking function. That is, it aims to output a ranked list of all reference subimages in descending order of similarity from a given query subimage. If necessary, each query subimage is resized to an appropriate size before being input to the self-localization system. This model is valid for most self-localization systems, which are based on image retrieval formulations.

Not only are rank lists output by self-localization algorithms, but some relevance scores for individual reference images (e.g., TF-IDF, conditional probabilities, maximal consistencies) are also output. However, it is not straightforward to interpret such relevance scores to LoC values within a detection-by-localization framework. We reserve this issue for future work.

Image-level and subimage-level rank values are available as output of any self-localization system. The BoVF-based self-localization systems first evaluate subimage-level similarities between each query subimage and each reference subimage, and then aggregate their similarity values to obtain the image-level similarity scores or rank lists. It is straightforward to output rank lists in the order of subimage-level similarity for individual query subimages.

C. Change Detection Model

The change detection model follows the standard formulation of image change detection [6]. Given a query image, it aims to estimate a pixel-wise LoC map of the image, which is then converted to a pixel-wise binary change mask. Its performance is evaluated in terms of 101-point (i.e., 0.00, 0.01, ···, 1.00) interpolated average precision (AP) [10].

III. OBJECT-LEVEL CHANGE DETECTION

This section provides detailed methods for object-level change detection via detection-by-localization (Fig. 2).

A. Scene Modeling

The scene modeling segments a given query/reference image to yield a pool of useful scene parts (in the form of bounding boxes) or subimages, so that the segmented subimages remain consistent between the reference and an unseen query images. This problem is referred to as “consistent part segmentation” [11]. This problem is significantly ill-posed, owning to differing FoVs, occlusions, and seasonal changes of object appearance. Even state-of-the-art segmentation algorithms are far from perfect, producing several false positive parts. We address this issue by hypothesizing a relatively large number of subimages and verifying them in the spirit of majority voting [3], as we explain in Section III-C.

We use both unsupervised and supervised segmentation techniques, inspired by our previous work [12]. Unsupervised segmentation techniques (e.g., BING [13]) provide category-independent object proposals, even for objects with unseen classes. We use a set of five pre-defined bounding boxes for every image. For an image with width $w$ and height $h$, these five bounding boxes are defined as $[w/3, 2w/3] \times [h/3, 2h/3]$, $[0, 2w/3] \times [0, 2h/3]$, $[w/3, w] \times [0, 2h/3]$, $[0, 2w/3] \times [h/3, h]$, and $[w/3, w] \times [h/3, h]$. Supervised segmentation techniques (e.g., YOLO [10]) provide more precise object regions supported by rich semantic information for objects with known classes. We use YOLO with threshold value of 0.05. It should be noted that whereas supervised techniques are trained on pre-defined object classes, it is often useful to propose unseen objects, having a visually similar appearance to pre-defined objects. Both supervised and unsupervised proposals are beneficial, because objects with both pre-defined and unseen classes can be changed objects.
B. Self-Localization

In [14], a novel AE-based ConvNet for loop-closure detection was presented. It was designed to be an unsupervised, convolutional AE network architecture, tailored for loop-closure, and amenable for efficient, robust place recognition. Its performance was evaluated via extensive comparison studies of the deep loop-closure model against state-of-the-art methods on different datasets. Additionally, the histogram of oriented gradients (HoG) was employed to compress images while preserving salient features and projective transformations (i.e., homography). In contrast, we are interested in the basic effectiveness of an AE-based self-localization, and we implement the convolutional AE in a rather simplified setting without using a HoG-based extension. Our AE consists of three convolutional layers and each convolutional kernel size is $3 \times 3$, employing max-pooling, batch-normalization and ReLU activation for each layer. Each input subimage is resized to $256 \times 256$, and mapped by the AE to a 16,388-dim feature vector.

C. Detection-by-Localization

Following our previous research in [11], we modeled individual scene parts belonging to different modalities, and adopted rank fusion techniques, derived from the field of multi modal information retrieval (MMR) [15]. MMR techniques are originally designed to deal with increasingly complex document collections, corresponding to subimage or part collections in our application domain and queries, consisting of not only text modalities but also non-textual modalities such as visual words in image retrieval, geo-tags, user rate, etc. More formally, they aim to fuse multiple retrieval results and queries from different modalities into a single ranked list to make a final decision. Unsupervised approaches requiring no training data are desirable, as explained in Section I. Such unsupervised MMR approaches are broadly classified into two categories: early fusion [16] and late fusion [17]. Early fusion aims to fuse multiple queries from multiple modalities at the level of the input feature descriptor. This approach is advantageous in exploring correlation between multiple data modalities. However, it requires multiple queries to be converted to the same format prior to fusion. This is not applicable to a wide range of self-localization applications. Late fusion aims to fuse multiple queries at the output decision level (e.g., the level of relevance score or rank list). This approach is further divided into score fusion [18] and rank fusion [17]. Score fusions rely on the so-called raw-score-merging hypothesis and fuses relevance scores from multiple queries. However, as a key limitation, it requires each modality’s retrieval to output a relevance score, significantly limiting its application. However, rank fusion aims to fuse rank values from multiple queries. As a key advantage, this approach requires only ranked lists (commonly output by self-localization), and importantly it can outperform the score fusion approach in previous applications [19].

Based on these considerations, we chose the rank fusion approach as our basis. In our previous study, the rank fusion approach was explored in a different context of image-level self-localization [11]. The current study is based on this previous method, but with key differences:

1) The previous study aimed at image-level ranking, whereas the current study aims to obtain subimage-level pixel-wise rank values.

2) The previous method took as input non-overlapping query subimages (from color-based segmentation), whereas the current study takes relatively large numbers of overlapping query subimages (from unsupervised/supervised object proposals).

To address this issue, we must deal with a novel task of pixel-wise rank fusion. Considering the new setting where each pixel may belong to multiple subimages (referred to as qBBs) as in Fig. 2, we fuse them all to obtain the rank list for each pixel. Because the number of such qBB is different among different pixels, it is not straightforward to compare the fused rank list between them. Our previous ranking method in [11] assumed that the number of rank lists was pre-defined and fixed. Thus, it was not directly applicable to the current problem. We now address this issue by introducing a new parameter, $N(i)$, the number of times a document appear in the rank lists, as suggested in [20]. The modified rule for rank fusion takes the form:

$$R(i) = N(i) \times \sum_{k=1}^{N(i)} \frac{1}{R_k(i)},$$

(1)

where $N(i)$ is the number of the $i$-th document appearing in the rank lists, and $R_k(i)$ is a rank value (of the ground-truth reference image) assigned by the $k$-th query’s retrieval. If $N(i)$ is a constant, the function $R(i)$ reduces to the previous fusion rule. Our algorithm takes as input a collection of query bounding boxes (qBBs), and performs the following steps:

1) Each of qBBs is checked, and if they overlap, the intersection areas are computed and registered as a new qBB; This process is repeated until no new qBBs are found.

2) For each $i$-th qBB, the number of overlaps, $N(i)$, is computed and related ranked lists are assigned, and then, the assigned ranked lists are fused by the function $R(i)$.

IV. Experiments

To test the performance of the proposed method, we performed extensive experiments on change detection, using the publicly available NCLT (North Campus Long-Term) dataset [21]. For each test data, we compared our pixel-wise rank
**TABLE I**

**PERFORMANCE RESULTS**

|               | AP     | RoC<sub>max</sub> |
|---------------|--------|------------------|
|               | 0.01   | 0.02  | 0.03  | 0.04  | 0.05  |
| rank fusion   | 0.74   | 0.73  | 0.72  | 0.72  | 0.72  |
| rank fusion (≤ 2) | 0.72   | 0.74  | 0.73  | 0.74  | 0.72  |
| rank fusion (≤ 3) | 0.72   | 0.74  | 0.73  | 0.74  | 0.74  |
| rank w/o fusion | 0.69   | 0.68  | 0.68  | 0.68  | 0.69  |
| score max     | 0.79   | 0.77  | 0.76  | 0.76  | 0.77  |
| score sum     | 0.73   | 0.71  | 0.70  | 0.70  | 0.71  |

The NCLT dataset is a large-scale, long-term autonomy dataset for robotics research collected at the University of Michigan’s North Campus by a Segway vehicle robotic platform. The data we used in the research includes view image sequences along vehicle’s trajectories acquired by the front facing camera of the Ladybug3 with GPS. From the viewpoint of change detection benchmark, the NCLT dataset has desirable properties:

1) It involves both indoor and outdoor change events during seamless indoor and outdoor navigations of the Segway robot.

2) It contains not only typical changed objects such as cars and pedestrians, but also various kinds of changes such as building construction, construction machines, posters, tables and whiteboards with wheels.

3) It has been recently widely used in robotics communities as experimental benchmarks for various tasks, such as self-localization [23].

In the current study, we use four datasets “2012/1/22”, “2012/3/31”, “2012/8/4”, and “2012/11/17” (referred to as WI, SP, SU, AU) collected across different four seasons. Fig. 3 shows the experimental environment and examples of changed objects in the dataset.

We formulated change detection as a binary classification problem. Each test sample consists of the pairing of two images from two different seasons, referred to as query and reference (or background) images. To create a test set, each image is viewed as a query image candidate and is paired with a reference image with a nearest neighbor viewpoint, forcing it to belong to a different season whose viewing angle must be nearer than 45 degree from that of the query image. The test set consists of positive and negative sets. A positive set consists of positive image pair samples, whose query image contains changes with respect to the counterpart reference image. A negative set consists of those image pairs with no change. For the positive set, these changed objects are manually annotated in the form of bounding boxes. The total number of positive and negative samples are 1,054 and 4,188. Image size is 1,232×1,616. Note that successive frames in the NCLT dataset often contain identical and visually similar changed objects in successive frames. To make our test set contain various changed fusion method with score fusion methods, which rely on the availability of relevance scores. Note that this score fusion method can be viewed as an adaptation of previous non-ranking-based (anomaly-based) change detection methods [22] to our problem domain. We used it to evaluate the level of achievement of the proposed method.
objects, we sampled at most one query image from such successive frames that contain such a visually very similar object. Thus, we obtained 1,054 positive image pairs from the $4 \times 3$ season pairs. Fig. 3 shows the robot’s viewpoint trajectories and representative changed objects in the test set.

For the test system, the visual appearance of each subimage is described by the AE as a feature vector. Feature vectors are indexed and retrieved by a nearest neighbor engine. Prior to the indexing, every feature vector is L2 normalized. For the nearest neighbor search, we employed two different object proposal methods (i.e., supervised and unsupervised methods). Most time consuming part of the test system are YOLO-based object proposal and AE-based part feature extraction, which were 11.1 msec per image and 0.9 msec per subimage (Geforce GTX Titan). The number of subimages per image was $15.9 \pm 3.0$ (mean±std), and thus total time cost was $11.1 + 0.9 \times 15.9 = 25.41$ msec.

Note that the length of the rank list can differ among query subimages. Therefore, raw rank values are not comparable among different queries. To address this issue, we normalize the raw rank values by the rank list length to the $[0,1]$-interval.

In our preliminary experiments, we also tested two variants of the proposed system:

1) One where the rank was not normalized, and
2) Another where reference subimages were sorted not in descending order but in ascending order of the LoC prior to the rank fusion.

We found that both variants did not work well and yielded significant performance drops (e.g., AP < 0.6). Moreover, we tested another variant where SIFT’s Harris-Laplace region and descriptor were used in place of qBBs and the AE descriptor, finding such a variant also does not work well. The reason might be that such a small Harris-Laplace region could not provide contextual and semantic information that are rich enough for change detection.

Difficulty indices play an important role in object detection literature standardizing benchmark datasets [24]. In the current study, we introduce two types of difficulty indexes: RoC and SoB. RoC (rate of change) is defined as the area of changed object (i.e., overlap region between the qBB and the ground-truth change qBB) normalized by the area of qBB. If RoC value is too small for a given qBB, it is difficult to detect such a small changed object. Thus, difficulty of change detection will increase. Effectiveness of this overlap-based difficulty index was verified in our previous study on self-localization [25]. SoB (size of bounding box) is defined as the area of qBB normalized by the area of image. If the SoB value is too small for a given qBB, such a subimage looks very dissimilar from the original non-cropped full image, and the probability of obtaining inappropriate self-localization result will be higher, because typical self-localization systems implicitly or explicitly assume that image size is the same or similar between the input query image and the corresponding reference image. We sampled a test query image set so that query images in the positive and negative sets respectively have restricted RoC values within their respective $[\text{RoC}^\text{min}_i, \text{RoC}^\text{max}_i]$ and $[\text{RoC}^\text{min}_i, \text{RoC}^\text{max}_i]$, and restricted SoB values within their respective $[\text{SoB}^\text{min}_i, \text{SoB}^\text{max}_i]$ and $[\text{SoB}^\text{min}_i, \text{SoB}^\text{max}_i]$. By default, we set $[\text{RoC}^\text{min}_i, \text{RoC}^\text{max}_i] = [0.9, 1.0]$, $[\text{RoC}^\text{min}_i, \text{RoC}^\text{max}_i] = [0.0, 0.05]$, $[\text{SoB}^\text{min}_i, \text{SoB}^\text{max}_i] = [0, 0.4]$, and $[\text{SoB}^\text{min}_i, \text{SoB}^\text{max}_i] = [0, 0.4]$. We create various test sets with different levels of difficulty by using $\text{RoC}$ as a difficulty index. More formally, the difficulty of a given test query image set is controlled by two thresholds $\text{RoC}^\text{min}$ and $\text{RoC}^\text{max}$. If, the $\text{RoC}$ value for a qBB exceeds the threshold of $\text{RoC}^\text{min}$ then the qBB is assigned the ground-truth change label of “change”. If, the $\text{RoC}$ value falls below the threshold of $\text{RoC}^\text{max}$ then the qBB is assigned the label of “no-change”. Other qBBs assigned neither “change” nor “no-change” are not considered as members of the test set. We created a collection of test sets with different levels of difficulty by changing these two threshold values.

We compared six different change detection schemes: “rank fusion”, “rank w/o fusion”, “rank fusion ($\leq 2$)”, “rank fusion ($\leq 3$)”, “score max”, and “score sum”. “Rank fusion” is the proposed algorithm. “Rank w/o fusion” is a variant that differs from the proposed algorithm because it does not fuse qBBs to create new qBBs. Therefore, it does not perform pixel-wise rank fusion. “Rank fusion ($\leq k$), $k=2, 3$” are motivated by the question “how many pixel-wise rank fusions are required to obtain optimal performance?”. Unlike the proposed algorithm, they perform pixel-wise rank fusion at most $k$-times per pixel. If there exists more than $k$ overlapping subimages for a pixel, $k$ subimages are randomly selected and used for that pixel. “Score max” and “score sum” are adaptations of “score max” and “score sum” strategies in MMR [15] per our application domain: the pixel-wise LoC map. These differ from the proposed algorithm, in that relevance scores are used in place of rank values (i.e., assuming the availability of relevance scores output by the self-localization system). For the relevance score, we use L2 distance of feature vectors. The “score max” fuses pixel-wise relevance score values $v_1, \cdots, v_k$ by fusion rule

$$v = \max_{i=1}^k v_i,$$

The “score sum” fuses them by fusion rule

$$v = \sum_{i=1}^k v_i.$$
Moreover, change detection performance was higher when pixel-wise rank fusion was used. The proposed rank-fusion scheme yields acceptable change detection performance? Table I shows the AP performance for different RoC max values. All the methods tested are not sensitive to the parameter values. Overall, the “score max” strategy yielded the best performance, whereas it requires the availability of relevance scores output by the self-localization system and relies on the raw-score-merging hypothesis. The proposed “rank fusion” strategy yielded comparable performance, despite it not requiring relevance scores. It clearly outperformed the “rank w/o fusion” strategy and other variants.

In summary, the proposed method successfully detected visually small objects from small (i.e., low-resolution) subimages. Moreover, change detection performance was higher when pixel-wise rank fusion was used. The proposed rank-fusion scheme is comparable to score-fusion schemes, which require the availability of relevance scores. Whereas the current study focuses on combining AE based feature vectors with pixel-wise rank fusion, in a future work, we plan to explore other self-localization schemes, such as different features (e.g., hand-crafted features), and heterogeneous self-localization systems (e.g., particle filters).

V. CONCLUSIONS AND FUTURE WORKS

We presented a new detection-by-localization method for object-level change detection using a ranking-based self-localization model, a novel model to evaluate the likelihood-of-change (LoC) of a given query subimage, and unsupervised rank fusion. The ranking-based model did not require training data, nor did it rely on raw-score-merging hypotheses. We consider the proposed unsupervised framework as beneficial to many visual SLAM systems, because such an unsupervised model can generalize it to a broad class of self-localization systems. Future work will investigate ways to use computational resources of different self-localization systems for large scale change detection and map maintenance applications. We will analyze its asymptotical behavior when the number of map databases and self-localization engines increases.

REFERENCES

[1] W. Churchill and P. Newman, “Experience-based navigation for long-term localisation,” The International Journal of Robotics Research, vol. 32, no. 14, pp. 1645–1661, 2013.
[2] J. Sivic and A. Zisserman, “Video google: A text retrieval approach to object matching in videos,” in null. IEEE, 2003, p. 1470.
[3] Y. Jiang, J. Meng, and J. Yuan, “Randomized visual phrases for object search,” in Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on. IEEE, 2012, pp. 3100–3107.
[4] P. Drews, P. Núñez, R. Rocha, M. Campos, and J. Dias, “Novelty detection and 3d shape retrieval using superquadrics and multi-scale sampling for autonomous mobile robots,” in Robotics and Automation (ICRA), 2010 IEEE International Conference on. IEEE, 2010, pp. 3635–3640.
[5] R. Finman, T. Whelan, M. Kaess, and J. J. Leonard, “Toward lifelong object segmentation from change detection in dense rgb-d maps,” in Mobile Robots (ECMR), 2013 European Conference on. IEEE, 2013, pp. 178–185.
[6] R. J. Radke, S. Andrea, O. Al-Kofahi, and B. Roysam, “Image change detection algorithms: a systematic survey,” IEEE transactions on image processing, vol. 14, no. 3, pp. 294–307, 2005.
[7] J. Im, J. Jensen, and J. Tullis, “Object-based change detection using correlation image analysis and image segmentation,” International Journal of Remote Sensing, vol. 29, no. 2, pp. 399–423, 2008.
[8] S. Khan, X. He, P. Porikli, M. Bennamoun, F. Sohel, and R. Togneri, “Learning deep structured network for weakly supervised change detection,” in Proceedings of the 26th International Joint Conference on Artificial Intelligence. AAAI Press, 2017, pp. 2008–2015.
[9] E. Garcia-Fidalgo and A. Ortiz, “bow-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, Oct 2018.
[10] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” CoRR, vol. abs/1612.08242, 2016. [Online]. Available: http://arxiv.org/abs/1612.08242
[11] K. Tanaka, “Unsupervised part-based scene modeling for visual robot localization,” in Robotics and Automation (icRA), 2015 IEEE International Conference on. IEEE, 2015, pp. 6359–6365.
[12] T. Sugimoto, K. Tanaka, and K. Yamaguchi, “Leveraging object proposals for object-level change detection,” in Intelligent Vehicle Symposium, IEEE, IEEE, 2018.
[13] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, “Bing: Binarized normed gradients for objectness estimation at 300fps,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 3286–3293.
[14] N. Merrill and G. Huang, “Lightweight unsupervised deep loop closure,” in Proc. of Robotics: Science and Systems (RSS), Pittsburgh, PA, Jun. 2018.
[15] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanahalli, “Multimodal fusion for multimedia analysis: a survey,” Multimedia systems, vol. 16, no. 6, pp. 345–379, 2010.
[16] R. Yan and A. G. Hauptmann, “Probabilistic models for combining diverse knowledge sources in multimedia retrieval,” Ph.D. dissertation, Carnegie Mellon University, Language Technologies Institute, School of Computer Science, 2006.
[17] G. V. Cormack, C. L. Clarke, and S. Buettcher, “Reciprocal rank fusion outperforms condorced and individual rank learning methods,” in Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, 2009, pp. 758–759.
[18] M. Montague and J. A. Aslam, “Condorcet fusion for improved retrieval,” in Proceedings of the eleventh international conference on Information and knowledge management, 2002, pp. 538–548.
[19] D. F. Hsu and I. Taks, “Comparing rank and score combination methods for data fusion in information retrieval,” Information retrieval, vol. 8, no. 3, pp. 449–480, 2005.
[20] M. Imhof and M. Braschler, “A study of untrained models for multimodal information retrieval,” Information Retrieval Journal, vol. 21, no. 1, pp. 81–106, 2018.
[21] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, “University of michigan north campus long-term vision and lidar dataset,” The International Journal of Robotics Research, p. 0278364915614638, 2015.
[22] P. Christiansen, L. N. Nielsen, K. A. Steen, R. N. Jorgensen, and H. Karstoft, “Deeponomaly: Combining background subtraction and deep learning for detecting obstacles and anomalies in an agricultural field,” Sensors, vol. 16, no. 11, p. 1904, 2016.
[23] J. G. Mangelson, D. Dominic, R. M. Eustice, and R. Vasudevan, “Pairwise consistent measurement set maximization for robust multi-robot map merging,” in Proceedings of the IEEE International Conference on Robotics and Automation, May 2018, pp. 1–8.
[24] F. Radenovic, A. Iscen, G. Toliás, Y. Avrithis, and O. Chum, “Revisiting oxford and paris: Large-scale image retrieval benchmarking,” in IEEE Computer Vision and Pattern Recognition Conference, 2018.

[25] K. Tanaka, “Self-localization from images with small overlap,” in Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on. IEEE, 2016, pp. 4497–4504.