International Journal of Control, Automation and Systems 17(X) (2019) 1-13
http://dx.doi.org/10.1007/s12555-018-0790-6
ISSN:1598-6446 eISSN:2005-4092
http://www.springer.com/12555

Bag of Sampled Words: A Sampling-based Strategy for Fast and Accurate Visual Place Recognition in Changing Environments

Sang Jun Lee and Sung Soo Hwang*

Abstract: In the field of visual place recognition, a variety of methods using Visual Bag of Words has been suggested to cope with environmental change. This paper presents a sampling-based method which improves the speed and the accuracy of the existing Visual Bag of Words models. We first propose sampling of image features considering their density to speed up the quantization step. By using samples, a more accurate but rather slow ranking procedure is feasible. Thus, we also propose a ranking procedure which utilizes spatial information of samples. Lastly, a coarse and fine approach-based refinement method is proposed which increases the accuracy of the system by iteratively updating the similarity between images. The experimental results show that the proposed method improves the performance of the existing Visual Bag of Words models in terms of speed and accuracy.

Keywords: Bag of words, image retrieval, visual place recognition.

1. INTRODUCTION

Visual-based place recognition, which started from image retrieval, has been studied for navigation and localization of robots in Simultaneously Localization And Mapping (SLAM) system. Several methods were based on Bag-of-Visual-Words (BoW) model [1, 2]. Other methods tried to use binary features to increase speed [3,4], and some suggested special data structures to build a codebook more efficiently [5–11]. A relaxed BoW model in terms of quantization, called soft assignment, was also suggested in [12]. Some method tried to rank the similar images using PCA in [13]. Some methods tried to use multi-feature detectors and descriptors [14] or another representation, called VLAD [15, 16] and Fisher Vector [17] to exploit the information of orientation of the vector quantized.

The applications using SLAM system are usually faced with changes in lighting, seasons, occlusion, and appearance. Hence, visual-based place recognition should be robust to those changes. Up to now, there have been the attempts to integrate several kinds of features to improve the accuracy of place recognition with environmental change [14,18,19]. They generate a robust BoW model by i) generating virtual views of a query image, and ii) extracting various kinds of features from these images. However, the computational cost of this method is too expensive, especially in quantizing the extracted features into BoW representation. It is caused by the fact that it normally extracts too much features from images. This can be a big disadvantage to be used in practical applications.

In this paper, we propose a novel method which can improve the existing BoW models in terms of speed and accuracy. Similar to previous approaches, the proposed method also generates virtual views to handle viewpoint changes, and extracts features by feature detectors and descriptors designed to be illumination-invariant. The main differences between the proposed method and the existing methods are as follows. First, we speed up the system by utilizing sampled features only instead of utilizing all extracted features. We present two kinds of sampling methods and compare the performance of the system depending on the sampling methods. One may argue that the accuracy of the proposed system can be degraded because only some of features are used. We overcome the degradation of the system by introducing a novel ranking procedure. Compared to other methods using TF-IDF [7], it uses spatial information, which produces results that are more accurate. It utilizes geometric verification, which examines the relation of two views in terms of epipolar geometry. The proposed ranking procedure is slower than the conventional ranking procedure. However, since we use samples of features for place recognition, the total speed of the system is not declined. Furthermore, we improve the accuracy of the system by utilizing the proposed refinement method. It iteratively updates the probability distribution of ranking list. Assuming that the target images are se-

Manuscript received October 31, 2018; revised March 21, 2019; accepted May 29, 2019. Recommended by Associate Editor Kang-Hyun Jo under the direction of Editor Euntai Kim. This journal was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2016R1D1A3B03934808).

Sang Jun Lee and Sung Soo Hwang are with the School of Computer Science and Engineering, Handong Global University, 558, Handong-ro, Heunghae-eup, Buk-gu, Pohang-si, KS010, Korea (e-mails: eowjd4@naver.com, sshwang@handong.edu).
* Corresponding author.

Springer

quentially captured from any vehicles, each iteration makes the target image to have higher probability by weighting the adjacent images. This method is done fast because the updating is only performed in ranking list.

In summary, the contributions of this paper are: i) utilization of sampled features for place recognition and two ways of feature sampling, ii) direct scoring of feature similarity by geometric verification, and iii) fast and iterative similarity refinement. By combining aforementioned three features, the proposed system can improve the existing BoW models in terms of speed and accuracy. Since the proposed method is based on BoW, yet utilizes sampled features, we name the proposed method *Bag of Sampled Words(BoSW)*.

This paper discusses related work in Section 2, and the detailed description of the above method in Section 3. We report our study of parameters and compare the performance with the existing method in Section 4. In Section 5, we discuss the conclusions.

2. RELATED WORK

2.1. Bag of words

BoW is one of the models for image retrieval, and BoW generation process can be divided into an off-line stage, an on-line stage, and a searching stage [1]. In the off-line stage, many features are extracted from each of images in the database. The extracted features of each image are then represented as a vector through the quantization process. The quantization process assigns the nearest index of a visual vocabulary to each feature. The on-line stage performs the same way for a query image. In the searching stage, the similarity between the representation of a vector of a query image and the representation of a vector of each of images in database is calculated as a score. Each scores is ranked to Top-N images, where the user assigns the N.

The visual vocabularies are learned by k-means [7] which is an unsupervised learning method. The hierarchical k-means (HKM) [6] was proposed to reduce the computational complexity. Approximate k-means (AKM) [5] uses a randomized k-d tree instead of using HKM to speed up with an approximate nearest neighbor search. For the quantization, previous methods usually use the k-d tree structure [8] for fast search. Best-bin-first modification [11] and randomized k-d forest [9, 10] are suggested to reduce the time complexity. The representation of a vector is converted to an inverted index file format [1, 20]. TF-IDF [7] scoring method is used to score the similarity. However, since the spatial information is lost in this score, re-ranking procedure using spatial information after ranking procedure is suggested in [5].

2.2. Feature extraction

Various types of handcraft features can be used for Bag of words model. The types can be divided into two categories as local feature descriptors, and global feature descriptors [2]. The local feature descriptors first detect the local features and describe each feature point, while global feature descriptors describe whole image. Bag of words model is based on the utilization of local feature descriptors.

Mishkin *et al.* has compared in [19] the local feature detectors for matching in extreme situations such as large baseline movement, illumination change, appearance, and occlusion. They reported that MSER [21] and Hessian-Affine [22] multiple feature detectors have the best performance for matching of image pairs. As for descriptor, [18] reported that the use of Root-SIFT (R-SIFT) and Half Root-SIFT (HR-SIFT) as multiple descriptors has the best performance for extreme situations.

Features detected by the handcraft method have disadvantages that they cannot utilize the semantic information of the image. Therefore, learning based feature extractor is possible as suggested in [23]. However, since the learning-based feature detectors require query images as training data to increase the accuracy, it cannot always define or list up the images for retrieval in advance.

2.3. Previous place recognition methods

In SLAM, BoW is widely used for localization [24]. There are several BoW methods such as DBoW [3] in ORB-SLAM [25] and G-SLAM [26], and FABMAP [4] in LSD-SLAM [27]. They use binary descriptors, i.e., ORB [28] and SURF [29], which can speed up the system for the real-time operation. These two methods have been proven to be performed well with high accuracy. However, in practice, place recognition methods using these features do not work well between images with different illuminations and with wide viewpoint change.

To address several changing environments, Mishkin *et al.* [14] suggested a method called Wide Baseline Stereo Generalization + BoW (i.e., WxBS+BoW). This method generates synthesis views of a query image for viewpoint-invariant image matching and extracts a lot of features from those images. This method outperforms the previous BoW models in terms of accuracy on challenging environments [30]. However, the computational cost is too expensive because this method requires big vocabularies (four 1M vocabularies), and the quantization step spends too many time to search the nearest neighbor in large vocabularies. Moreover, because the geometric verification is performed by using RANSAC-like method [31], the speed is too degraded.

Currently, VLAD-based place recognition method [16] is proposed as lightweight and viewpoint-invariant method. The VLAD used in this method aggregates features that have same position by BoW model, and increases performance by exhaustive feature matching approach, and reduces the dimension of the representation by data-independent dimensionality reduction. However,

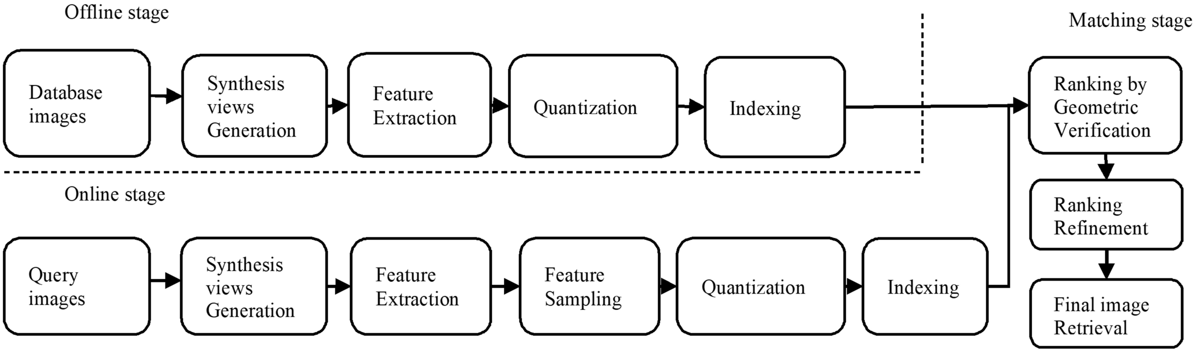


Fig. 1. System overview of the proposed bag of sampled words model.

this method has shown a trade-off between computation time and performance.

3. METHOD

3.1. System overview

The overview of the proposed BoSW is shown in Fig. 1. Similar to the conventional BoW model, the system is dividied into an offline stage and an online stage. In the offline stage, the quantization step is performed for each of images in database. Each image generates synthesized views and extracts features for all these views. These features are quantized to represent a vector of each image by finding the closest vocabulary from a code book. The code book is pre-trained by using same feature descriptors. The quantized features are indexed by using inverted file format. In the online stage, syntheses views are generated, and features are extracted similar to the offline stage. However, unlike the offline stage, features sampled by the proposed sampling method are quantized and indexed rather than using whole features.

In the matching stage, a query image is matched with each image in database. Each match calculates a similarity using geometric verification, and images are ranked according to the proposed similarity measure. Then, the proposed refinement method updates the distribution of the similarity with fast and iterative refinement and re-rank the similarities. Finally, the top image in the ranking list is detected as the final image.

3.2. Synthesis view generation

As suggested in [32], we generate synthesized views for all images from database as well as query images. For the comprehension of this paper, we will briefly explain how to generate synthesizes views.

An image can be transformed into parameterized homography transformation. An affine transformation matrix A can parameterize the homography transformation by first order Taylor expansion [19]. This A can be decomposed by Singular Value Decomposition (SVD) as follows:

$$\begin{aligned}\mathbf{A} =& \mathrm{H}_\lambda \mathrm{R}_1(\psi)\mathrm{T}_t\mathrm{R}_2(\phi)\\ =&\lambda\begin{pmatrix}\cos\psi & -\sin\psi\\ \sin\psi & \cos\psi\end{pmatrix}\begin{pmatrix}t & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}\cos\phi & -\sin\phi\\ \sin\phi & \cos\phi\end{pmatrix},\end{aligned} \tag{1}$$

where $\lambda > 0$, $\mathrm{R}_1(\cdot)$ and $\mathrm{R}_2(\cdot)$ are rotation matrices, and T_t is a diagonal matrix with $t > 1$. As t is an absolute tilt, the latitude is $\arccos(1/t)$. ϕ is the longitude, and ψ is the rotation of the camera about the optical axis. Assuming that ψ is fixed to the gravity assumption, the parameterization is t and ϕ, and optionally scale factor. Then, the synthesized view generation process is performed in Gaussian scale-space by convolution of a Gaussian filter multiplied by a down sampling factor s, rotated by t and ϕ.

3.3. Feature extraction

As mentioned in Section 2, the combination of MSER and Hessian-Affine was reported as the best feature detectors in challenging situations. While MSER is the robust feature detector at the appearance of the many structured objects such as building [21], Hessian-Affine is relatively robust to nature scenes [22]. It means that the Hessian-Affine tends to extract features in the nature scene. However, extracting features from the nature scene must be avoided because the nature scene itself is prone to be changed. Moreover, since the trees will change their colour and size as they grow up, they may give defects at long term matching. It is also difficult to match the prominent images for place recognition because they have high frequency features that are repeated in same pattern. Therefore, we assume that at least one building or object should be existed in the scene for place recognition. And according to this assumption, we extract features using the MSER detector suggested in [19].

3.4. Feature sampling

If the number of detected features is large, the time complexity of the system is increased. Therefore, rather

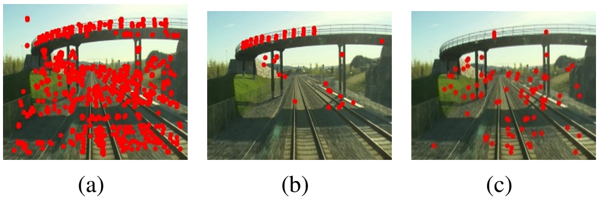


Fig. 2. An example of sampling ($S = 100$). (a) The detected whole features. (b) The samples using random sampling. (c) The samples using density-based sampling.

than using the whole extracted features, we utilize samples of features. Samples should represent a population of detected features without bias. One possible way of sampling is random sampling [33]. The random sampling samples each individual that has same probability of being chosen during the sampling process. It easily samples without any prior knowledge of the distribution of the population.

Even though random sampling has its own advantages, we can consider the distribution of the features if we know that in advance. Such sampling is possible to consider local density of features. It can be proceeded as follows. First, the distribution of features is estimated by multi variate kernel density estimation [34] according to the location of features. In this case, we use the Epanechnikov distribution [35] as the kernel to estimate density, which is known as the optimal distribution in terms of mean squared error. We use an approximate mean squared error to estimate the optimal bandwidth for density estimation.

Let $f(x)$ is the distribution of features estimated by multi-variate kernel density estimation. The sampling considering $f(x)$ distribution can be performed similar to Rejection sampling [36]. For this purpose, $f(x)$ is divided by the largest value in $f(x)$ for normalization as follows:

$$f'(x) = \frac{f(x)}{\max(f(x))}, \tag{2}$$

where $f'(x)$ is the normalized distribution of the $f(x)$. For a random sample of a feature X, the sample is accepted if u $< f'(x = X)$ where the u is a value sampled from uniform distribution $\mathcal{U}(0,1)$. Otherwise, the sample is rejected. The sampling of this process is continued until the total number of the accepted samples become S.

The samples of features by using random sampling and using the density-based sampling are shown in Fig. 2(c). In Section 4.2, the experimental results will show the performance comparison between random sampling and density-based sampling.

3.5. Direct ranking by geometric verification

The speed of the quantization using samples of features is T/S times faster than the one using the whole features where T is the number of whole features and S is the number of samples of features. The smaller the number of S, the faster the speed of the system. However, the scoring process using samples by the conventional TF-IDF method may be inaccurate because the samples has limited information in comparison with the all features. Fortunately, we can directly calculate each of scores using spatial information. It is possible by the fact that sampling of features can speed up the process. This direct ranking method using spatial information is robust to the matching of large viewpoint movement owing to ranking using geometry information. Hence, we can compensate the effect of utilizing small number of features.

The direct ranking method is done by performing the geometric verification [37], and it is performed by calculating the Fundamental matrix [38] between a query image and a target image in database. It means that the geometric verification checks the relation of epipolar geometry because if the target image is true positive, the camera pose between the query image and the target image is related by rigid body transformation.

Therefore, to check this, we first find the corresponding pair points of samples between a query image and target image. The location information of them can be easily taken from the indices of the quantized inverted files. We use Lo-RANSAC [39] to calculate the fundamental matrix. To reduce the time complexity to calculate the fundamental matrix, duplicated matching pairs are removed and a local affine frame check suggested in [18] is performed to remove the outlier matching pairs. The samples of a query image are mapped to a target image using the fundamental matrix.

Finally, the distance error on Euclidean space between the mapped samples and the corresponding samples of target image is calculated as a score. For all target image in database, all the scores after performing the geometric verification is sorted by the ascending order to place an image having lower sum of distance error on the top in the ranking list.

Even if we use S samples in a query image, the number of corresponding pairs is different from each of images in database. To normalize the number of corresponding pairs, we normalize the corresponding pairs number $\mathcal{M}_x$ to 100 (i.e., $\mathcal{M}_x$*100/$\mathcal{M}_x = 100$, where $x \in$ images in database). Therefore, we get the scores about target image x as follows:

$$score(x) = d_x * \frac{100}{\mathcal{M}_x}, \tag{3}$$

where d_x is the un-normalized distance error of the image x. Note that this scoring step is performed only for the images in $\mathcal{M}_x > M$ where M is the threshold for the number of matching pairs. This strategy improves the speed of ranking procedure. After sorting the scores, only the top N items are stored in the shortlist L.

3.6. Ranking refinement

The loss of information caused by utilizing samples of features is further compensated by the supplement of additional samples under the Law of large numbers [24]. The first method for the supplement is to increase the number of samples itself. However, this eventually takes lots of time to compute the fundamental matrix for geometric verification in the RANSAC-like method. Secondly, we can sample repeatedly to support the samples in the same query image updating the probability distribution of score more precisely. This can improve accuracy. Therefore, we use the refinement method that the ranking of score updates using the coarse-fine based iterative method.

Since the estimated density of probability distribution of scores have Multi-modes, we update that the high mode be higher, and the low mode be lower in each iteration. To do this, we repeatedly perform image retrieval in the same query image and multiply the estimated density to the previous one. Eventually, it will converge to τ modes that stand out, and then re-scores the found modes by the each of probability values.

The implementation steps of this method are as follows. First, given the $score(x)$, we need to convert $score(x)$ to probability distribution by normalization. Because the $score(x)$ is sorted by ascending order, we make a probability distribution $p(x)$ by applying the inverse of $score(x)$ so that the smaller score values have higher probability values as:

$$p(x) = \frac{1}{\mathrm{Z}} \cdot \frac{1}{score(x)}, \quad x \in \text{images in database}, \tag{4}$$

where Z is a partition function that allows the sum of $p(x)$ to be 1. Then, the probability distribution $p(x)$ is convolved with a Gaussian filter $g(x)$ with a standard deviation of σ to estimate the probability distribution $q(x)$ as follows:

$$q(x) = (p * g)(x), \tag{5}$$

where $*$ is a convolution operator. The reason for operating the convolution is to increase the probability among adjacent images. If the adjacent images are existed in the shortlist, the possibility of that the one of images is the true positive image increases because the images in database are sequentially made by SLAM system.

The next step is the updating step. Let the k is the number of iterations. The $q_k(x)$ indicates the probability distribution of a query image in each iteration k. When k is 0 as initial iteration, from next iteration each iteration multiplies the output probability distribution in k-th execution to the output probability distribution in $(k-1)$-th iteration. Thus, the equation of updating probability distribution for each iteration is following as:

$$\mathrm{Q}_0(x) = q_0(x),$$

$$\mathrm{Q}_1(x \mid q_1, \mathrm{Q}_0) = \frac{1}{\mathrm{Z}} \cdot q_1(x)\mathrm{Q}_0(x),$$

$$\vdots$$

$$\mathrm{Q}_k(x \mid q_k, \mathrm{Q}_{k-1}) = \frac{1}{\mathrm{Z}} \cdot q_k(x)\mathrm{Q}_{k-1}(x), \tag{6}$$

$$\approx \mathrm{Q}_k(x \mid q_k, q_{k-1}, \cdots, q_0) = \frac{1}{\mathrm{Z}} \cdot \prod_{i=0}^{k} q_i(x), \tag{7}$$

where x $\in$images in database, $\forall x \in \mathrm{Q}_{k-1}(x) > 0$. The Q_i ($i = 0$, ..., k) is a probability distribution for the final score at each iteration, and Z is a partition function that allows the sum of $\mathrm{Q}_i(x)$ to be 1. At this time, the update process in each execution is very quick because it is only performed for all images $x \in \mathrm{Q}_{k-1}(x) > 0$. The mode can be counted where $\mathrm{Q}_k(x-1)$ and $\mathrm{Q}_k(x+1)$ being smaller than $\mathrm{Q}_k(x)$. This updating procedure repeats until the number of modes becomes less than τ. The final image is detected by the highest probability value among the detected τ images. An example of the iterative refinement is shown as Fig. 3. As iteration is repeated, we can see that the probability distribution is updated as the elaborate modes to the true positive image.

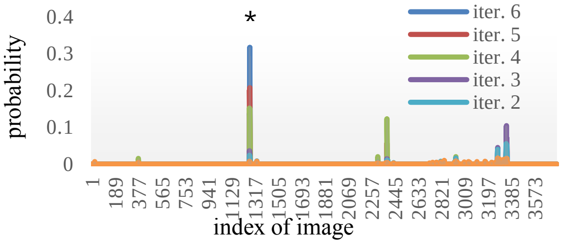


Fig. 3. An example of results of suggested refinement method. The $*$ is the true positive image.

4. EXPERIMENAL RESULTS

4.1. Environment setup

We implemented the proposed system on i-7 CPU and 8G ram with C++. We only used one thread in all the experiments. They were generated for two descriptors, HR-SIFT and R-SIFT. They were learned by AKM [5] and quantized into a randomized k-d tree. The parameter setting for the synthesis view generation was as follows: $\{s\} = \{1;\ 0.25;\ 0.125\}$, $\{t\} = \{1;\ 3;\ 6;\ 9\}$ and $\Delta\phi = 360^\circ/t$. Each syntheses view was created several times by the above parameters. We set parameters empirically $\tau = 3$ for the final candidate images and $N = 100$ for the shortlist. We used Gaussian filter for the convolution in refinement method with 1×5 normal distribution $N(0,1)$.

Prior to the experiments, we defined the recall, precision, localization rate and detection rate similar to [5,40] as shown in Table 1. Each measurement means that the

Table 1. Types of measurements.

Recall	$\frac{\text{the number of query images at least one in the shortlist}}{\text{the number of query images}}$
Precision	$\frac{\text{the number of query images at the top 1}}{\text{the number of query images at least one in the shortlist}}$
Localization	$\frac{\text{the number of query images at the top 1}}{\text{the number of query images}}$
Detection rate	$\frac{\text{the number of query images whose the number of matching pairs larger than M}}{\text{the number of query images}}$

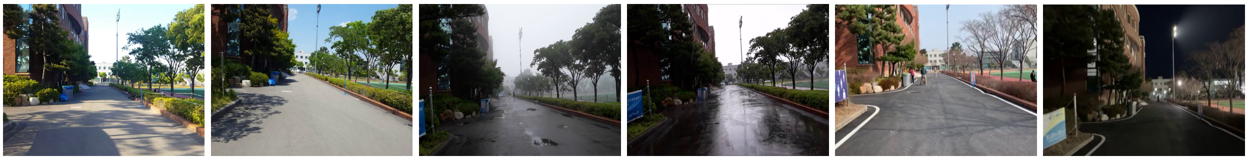

(a) 1 p.m. (b) 5 p.m. (c) fog. (d) rain. (e) 10 a.m.-winter. (f) night-winter.

Fig. 4. Self-collected dataset of campus. This dataset is composed of a variety of illumination changes, occlusions, viewpoint changes, and season changes.

recall is a measurement of the likelihood of a true positive image in the shortlist. The precision is a measurement of the likelihood how many the true positive image in the shortlist is top 1, and the localization rate is a measurement of detecting the true positive image as the top 1 from whole retrieval. Thus, the important thing is to find high localization rate. The detection rate means that how many the true positive image is detected even it is not in shortlist.

We tested the proposed method on the dataset *VPRiCE(Visual Place Recognition in Changing Environments)* [30] and a self-collected *Campus* dataset. The *VPRiCE* dataset consists of various images captured from a variety of vehicles such as trains, cars, buses, bicycles, pedestrians with different environments such as viewpoint change, occlusion, and illumination changes like day-night time change or seasonal change. This dataset is divided into two datasets, *memory* with 3756 images for database and *live* with 4022 images for queries. We created visual vocabularies using all images in *VPRiCE*, and categorized the images in both *live* and *memory* dataset according to the used vehicles with descriptions in Table 2. For the experiments, we selected 56 images in the proportion of the number of images according to the categories in *live* dataset. *Campus* dataset is composed of 12K images that have illumination changes, occlusions, viewpoint changes, and season changes. These images are taken from video by hand-held and are divided into several different situations such as 1 p.m., 5 p.m., fog, rain from summer, and 10 a.m., night from winter as shown in Fig. 4. We compare the proposed BoW models with existing BoW model using this dataset in Section 4.6.

Since the ground truth for each query image is not provided, the most similar ones are indexed and used as the ground truth. If the index of the searched image is equal to the ground truth, we labelled *OK*. If it is not the same place but similar to the ground truth image, we labelled *Good*. If there is no overlapping space with non-similarity, we labelled *Bad*. Only the images detected as *OK* are recognized as true positive image, and the recall, precision and localization rate are calculated by using this. We searched all the images in the *memory* dataset and recorded the results for each of categories in the each of experiments.

4.2. Comparison of sampling methods

We compared the performances between random sampling and density-based sampling. The test was performed on 10K vocabularies and $S = 100$ samples from *VPRiCE*. We calculated the mean and standard deviation of the number of matching pairs in each category with detection rate. The results are shown in Table 3.

The experimental results show that the density-based sampling is lower than the random sampling in terms of the number of matching pairs. However, the detection rate is higher. This means that the density-based sampling detects more samples which reflects the characteristics of each image (distribution of the features) without duplication. Therefore, it has fewer matching with negative images and higher matching with true positive images. However, high detection rate does not guarantee high precision and localization rate. In fact, precision and localization rate is more affected by the threshold M which is the threshold for the number of matching pairs than detection rate. Therefore, we compared precision and localization rate with respect to M in Section 4.4 by using both sampling methods.

Table 2. Descriptions regarding categories for the datasets.

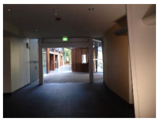
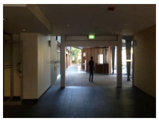
Category	An example in live dataset	An example in memory dataset	description
Catagory 1			➢ These pictures are taken from trains. ➢ Nature environments with few or no buildings. ➢ Large illumination change, low viewpoint change, few occlusions and few appearance changes.
Catagory 2			➢ These pictures are taken from pedestrians. ➢ Many buildings with few or no nature environments. ➢ Little illumination change, low viewpoint change, some occlusions and few appearance changes.
Catagory 3			➢ These pictures are taken from cars. ➢ Few buildings with some nature environments. ➢ Large illumination change, middle viewpoint change, few occlusions and middle appearance change.
Catagory 4			➢ These pictures are taken from buses. ➢ Many nature environments with few buildings. ➢ Little illumination change, large viewpoint change, many occlusions and large appearance change.
Catagory 5			➢ These pictures are IR (infrared) image taken from bikes. ➢ Some nature environments with few buildings. ➢ Large illumination change, large viewpoint change, few occlusions and little appearance change.

Table 3. Comparison of sampling methods.

	Density-based sampling		Random sampling	
	Number of matching pairs	Detection rate	Number of matching pairs	Detection rate
Category 1	52±14	0.467	**58**±20	**0.533**
Category 2	47±11	**1.000**	**55**±10	0.916
Category 3	68±20	**1.000**	**78**±26	0.857
Category 4	**69**±45	**0.778**	**69**±27	**0.778**
Category 5	61±43	**0.538**	**98**±38	0.308

4.3. Comparison of number of vocabularies

In general, BoW model utilizes 1M vocabularies. However, since we propose a new BoW model (i.e., BoSW model), we need to find the proper number of vocabularies. For this purpose, we calculated the quantization speed (q. s.), the searching speed (s. s.), recall, precision, and localization rate with respect to the number of vocabularies, 10K,100K, 1M. The quantization speed indicates the speed of quantization step, and the searching speed indicates the speed of ranking step. We used 100 samples for this experiment. At this time, we only used the R-SIFT descriptor and 1252 images in the *memory* dataset of *VPRiCE* by skipping the index of the image by three. Since the proposed system is influenced by the threshold M, we set the M for 10K to 30, for 100K to 10, for 1M to 1. The reason why the larger the number of the vocabularies, the smaller M is because the quantized representation of samples using the larger vocabularies is going to be sparser.

The experimental results are shown in Table 4. In Table 4, the first row indicates the number of vocabularies and the threshold M in parentheses. The experimental results show that the smaller the number of vocabularies, the faster the quantization speed of the samples. In addition, there is a tendency that the larger number of vocabularies has the more precise precision with high searching speed. However, the recall is low. Even though the precision and localization rate are important, we select 10K as the proper number of vocabularies because we can increase the precision and localization rate if the recall is high. In Table 4, 10K slots have larger recall than others do where the blank means that it detects nothing.

4.4. Parameter testing

In this part, we tested the performance of the proposed system depending on the number of samples and the

Table 4. Comparison of the number of vocabularies.

	10K (50)					100K (10)					1M (1)				
DB	q. s.	s. s.	Rec.	Prec.	Loc.	q. s.	s. s.	Rec.	Prec.	Loc.	q. s.	s. s.	Rec.	Prec.	Loc.
Catagory 1	0.50 ±0.01	1.36 ±1.64	0.267	0.750	0.200	8.43 ±0.17	0.74 ±0.70	0.267	1.000	0.267	134.11 ±7.07	0.26 ±0.04	0.133	1.000	0.133
Catagory 2	0.53 ±0.06	1.90 ±3.54	0.667	0.200	0.300	8.48 ±0.36	0.48 ±0.45	0.750	0.778	0.583	153.98 ±23.21	0.43 ±0.50	0.333	1.000	0.333
Catagory 3	0.51 ±0.01	11.64 ±10.90	0.429	0.000	0.000	8.63	0.38	0.143	1.000	0.143					
Catagory 4	0.52 ±0.02	12.75 ±8.04	0.556	0.000	0.000										

Table 5. Comparison between the number of samples and threshold of matching pairs of random sampling method.

DB	Method	f. q. s.	s. s.	Rec.	Prec.	Loc.	#sample	threshold
Catagory 1	Ref.	3.77 ± 1.56	9.19 ± 15.22	0.40	1.00	**0.40**	100	50
Catagory 2	Ref.	17.40 ± 12.47	34.46 ± 18.56	0.75	1.00	**0.75**	150	50
Catagory 3	Ref.	9.45 ± 9.23	54.64 ± 7.42	0.43	1.00	**0.43**	150	50
Catagory 4	Ref.	.35 ± 4.61	16.47 ± 14.40	48.24 ± 17.13	0.44	1.00	150	60
Catagory 5	-	-	-	-	-	-	-	-

Table 6. Comparison between the number of samples and threshold of matching pairs of density-based sampling method.

DB	Method	f. q. s.	s. s.	Rec.	Prec.	Loc.	sample	threshold
Catagory 1	Ref.	5.87 ± 3.28	18.66 ± 24.04	0.33	1.00	**0.33**	150	60
Catagory 2	Ref.	14.87 ± 10.73	32.09 ± 20.67	0.75	1.00	**0.75**	150	50
Catagory 3	Ref.	6.20 ± 4.59	16.69 ± 21.85	0.71	0.80	**0.57**	100	50
Catagory 4	Ref.	.35 ± 4.61	45.00 ± 17.66	0.33	0.67	**0.22**	150	60
Catagory 5	-	-	-	-	-	-	-	-

threshold M. For this purpose, we used 56 *live* images in *VPRiCE* as query images using HR-SIFT and R-SIFT descriptors with 10K vocabularies for 3756 *memory* dataset images. The threshold M was set to $\{30, 40, 50, 60, 70, 80, 90, 100\}$ and the number of samples S was set to $\{50, 100, 150, 200, 250, 300\}$. Table 5 and Table 6 show the best results of parameter testing as we changed the number of samples and the threshold M with respect to the random sampling and density-based sampling. Note that *f. q. s.* is the speed of feature extraction and quantization, *No.* is the method that the proposed refinement is not performed, and *ref.* is the method that the refinement is performed.

We also measured the time and accuracy for each category among those combinations between S and M. The unit of time is second and the average and variance of the time is calculated for each category. At this time, we stop image searching when the sum of *f. q. s.* and *s. s.* exceed 60 seconds and recognize that the query image is not found. The bold-type indicates best result.

In the experimental results, the combinations of using 100, 150 samples and setting the threshold value as 50, 60 were the best for both sampling methods in terms of localization rate. Therefore, we chose the combination of 150 samples and threshold value as 50 for random sampling, and 150 samples and threshold value as 60 for density-based sampling. We compared the proposed methods using these combinations with other methods in Section 4.5 and Section 4.6.

4.5. Comparisons on *campus* dataset

Next, we compared the proposed BoSW model with existing BoW model on *campus* dataset. Here, we tried to verify the improvement of existing method using the proposed system in terms of accuracy. Hence, we built the proposed model on top of the modified existing method, and the descriptor HRSIFT and RSIFT were equally used.

Table 7. Comparison with existing methods on capmus dataset.

BoSW + d.s. + ref. with 150 samples and 50 threshold						
Quary DB	1 p.m.	5 p.m.	Fog	Rain	10 a.m.	night
1 p.m.	-	**0.33**	**0.33**	**0.47**	**0.33**	**0.07**
5 p.m.	0.40	-	**0.33**	**0.47**	0.20	**0.20**
Fog	**0.40**	**0.33**	-	0.47	**0.20**	**0.07**
Rain	**0.20**	**0.27**	**0.27**	-	**0.20**	**0.20**
10 a.m.	**0.27**	**0.33**	**0.27**	**0.13**	-	**0.20**
night	**0.20**	**0.20**	0.13	0.13	**0.20**	-
BoW						
DB Quary	1 p.m.	5 p.m.	Fog	Rain	10 a.m.	night
1 p.m.	-	**0.33**	0.13	**0.47**	0.27	0.07
5 p.m.	**0.47**	-	0.13	0.40	**0.27**	0.00
Fog	0.33	0.13	-	**0.53**	0.13	**0.07**
Rain	**0.20**	0.20	**0.27**	-	**0.20**	0.00
10 a.m.	0.13	0.27	0.13	0.07	-	0.00
night	0.00	0.07	**0.13**	**0.13**	0.07	-

We modified the existing method by using 10K vocabulary and only MSER detector for efficient experiments. It is because it required too many computing powers if experiments were performed on 1M vocabulary and MSER with Hessian as it was suggested. For the experiment, we manually selected 10 landmarks in each situation, and performed image retrieval for all situations.

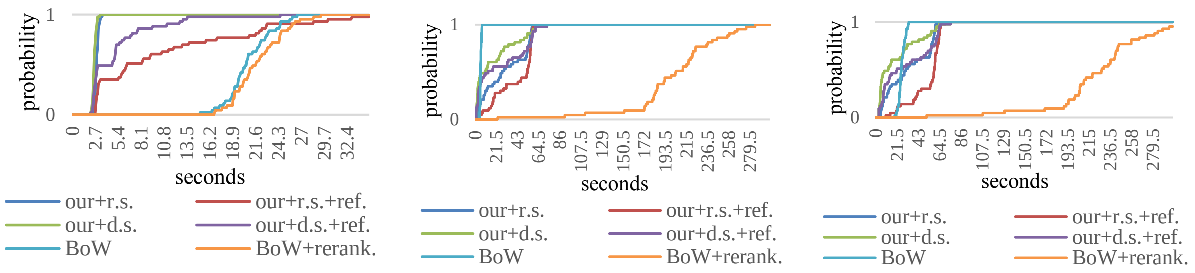
The experimental results are shown in Table 7. In Table 7, *BoW* refers to the method that uses just TF-IDF scoring of existing method [7], and *BoSW + d.s. + ref.* refers to the proposed method with density-based sampling and refinement. Using *BoSW+d.s.+ref.* with 150 samples and 50 as threshold value, we compared localization rate with *BoW* model. In most cases, localization rate in the proposed method outperforms the existing BoW model. Therefore, we can conclude that the proposed method can improve the performance of the existing BoW model.

4.6. Comparison with existing methods on *VPRiCE*

Here, we compared the proposed BoSW model with existing method using BoW model suggested in [14], and VLAD model suggested in [16]. The experimental results are shown in Table 8. In the Table 8, *BoW + rerank.* refers to the application of the re-ranking process after TF-IDF scoring. *BoSW + r.s.* refers to the proposed model using random sampling only, and applying refinement is *BoSW + r.s. + ref.* refers to the proposed model using random sampling and refinement. and *BoSW + d.s.* refers to the proposed model using density-based sampling without refinement, and *BoSW + d.s.+ ref.* refers to the proposed model using density-based sampling without refinement.

Table 8. Comparison with existing methods on VPRiCE dataset.

	f.q.s	s.s	Rec.	Prec.	Loc.
Method	Catagory 1				
BoW	20.35 ± 1.72	2.65 ± 0.29	**0.73**	0.36	0.27
BoW+rerank	20.82 ± 2.01	192.85 ± 20.32	0.67	0.30	0.20
BoSW+r.s.	2.81 ± 0.17	35.37 ± 18.72	0.40	0.67	0.26
BoSW+r.s.+ref.	12.2 ± 12.44	45.15 ± 15.69	0.40	0.83	0.33
BoSW+d.s.	2.63 ± 0.16	14.81 ± 15.69	0.33	**1.00**	0.33
BoSW +d.s.+ref.	5.87 ± 3.28	18.66 ± 24.04	0.4	**1.00**	**0.40**
VLAD [16]	0.9	0.17	0.6	0.67	**0.40**
	Catagory 2				
BoW	21.36 ± 3.59	2.41 ± 0.25	**1.00**	0.75	**0.75**
BoW+rerank	21.36 ± 3.59	189.23 ± 52.64	0.92	0.73	0.67
BoSW+r.s.	2.97 ± 0.17	22.27 ± 20.73	0.75	0.56	0.42
BoSW+r.s.+ref.	17.40 ± 12.47	34.46 ± 18.56	0.75	**1.00**	**0.75**
BoSW+d.s.	2.55 ± 0.27	15.16 ± 19.66	0.92	0.55	0.50
BoSW +d.s.+ref.	5.77 ± 3.47	17.96 ± 21.32	0.83	0.80	0.67
VLAD [16]	1.2	0.18	**1.00**	0.75	**0.75**
	Catagory 3				
BoW	22.36 ± 2.64	2.74 ± 0.30	**1.00**	0.00	0.00
BoW+rerank	24.50 ± 2.74	224.05 ± 27.32	**1.00**	0.29	0.29
BoSW+r.s.	3.02 ± 0.30	41.27 ± 22.2	0.43	0.00	0.00
BoSW+r.s.+ref.	9.45 ± 9.23	54.64 ± 7.42	0.43	**1.00**	**0.43**
BoSW+d.s.	2.66 ± 0.19	22.68 ± 19.29	0.71	0.40	0.29
BoSW +d.s.+ref.	7.18 ± 7.27	30.91 ± 21.53	0.71	0.60	**0.43**
VLAD [16]	1.1	0.17	0.86	0.50	**0.43**
	Catagory 4				
BoW	22.54 ± 2.34	2.88 ± 0.39	**0.56**	0.00	0.00
BoW+rerank	22.82 ± 2.12	237.16 ± 21.07	**0.56**	0.20	0.11
BoSW+r.s.	2.81 ± 0.16	45.95 ± 18.45	0.22	0.50	0.11
BoSW+r.s.+ref.	5.39 ± 3.39	52.88 ± 12.88	0.22	0.50	0.11
BoSW+d.s.	2.66 ± 0.15	28.64 ± 16.15	0.44	0.25	0.11
BoSW +d.s.+ref.	7.35 ± 4.61	45.00 ± 17.66	0.33	**0.67**	0.22
VLAD [16]	0.9	0.17	0.56	0.60	**0.33**
	Catagory 5				
BoW	21.23 ± 1.58	2.60 ± 0.20	**0.23**	0.00	0.00
BoW+rerank	26.07 ± 7.17	248.18 ± 41.78	**0.23**	0.00	0.00
BoSW+r.s.	2.86 ± 0.19	50.03 ± 18.56	0.08	0.00	0.00
BoSW+r.s.+ref.	5.98 ± 6.25	53.99 ± 11.12	0.00	0.00	0.00
BoSW+d.s.	2.70 ± 0.28	42.05 ± 21.20	0.15	0.00	0.00
BoSW +d.s.+ref.	3.47 ± 2.08	24.43 ± 20.62	0.15	0.00	0.00
VLAD [16]	0.87	0.17	0.00	0.00	0.00



(a) (b) (c)

Fig. 5. The graphs for measurement of speed. (a) The Cumulative histograms for feature extraction and quantization speed. (b) The Cumulative histograms for searching speed. (c) The Cumulative histograms for speed.

Each method used best parameters found in Section 4.4. And bold-type indicates the best value, and underline denotes the second best value.

In terms of the speed, the proposed methods show 10 times improvement of quantization speed of existing BoW model. The searching speed is usually faster than existing method that uses re-ranking procedure. We also express the speed of systems graphically for more intuitive comparison in Fig. 5. Fig. 5(a) is a cumulative histogram comparing features extraction and quantization speed. Fig. 5(b) is a cumulative histogram of the searching speed, and Fig. 5(c) is a cumulative histogram of the total time. As the results, the proposed method was the fastest in quantization speed. Searching time and total time are slower than *BoW* method, but much faster than *BoW* + *rerank.* method. It means that the *BoW* + *rerank.* is constantly slow because that uses re-ranking step, but the proposed method is fast even we rank using the robust ranking method. At the quantization speed, the method using refinement takes longer than non-refinement method. This is because it performs quantization repeatedly. However, since the quantization speed itself is fast, the repetition is faster than the *BoW* + *rerank.* method.

In terms of accuracy, the proposed method is the best on precision rate in most cases. VLAD is the best on localization rate, but the proposed method is also the best or the second best. The reason why the localization rate of VLAD method is more accurate than the proposed methods' one is that the recall of VLAD is higher than the proposed methods' recall. The higher recall, the more candidates for localization be detected.

4.7. Discussion

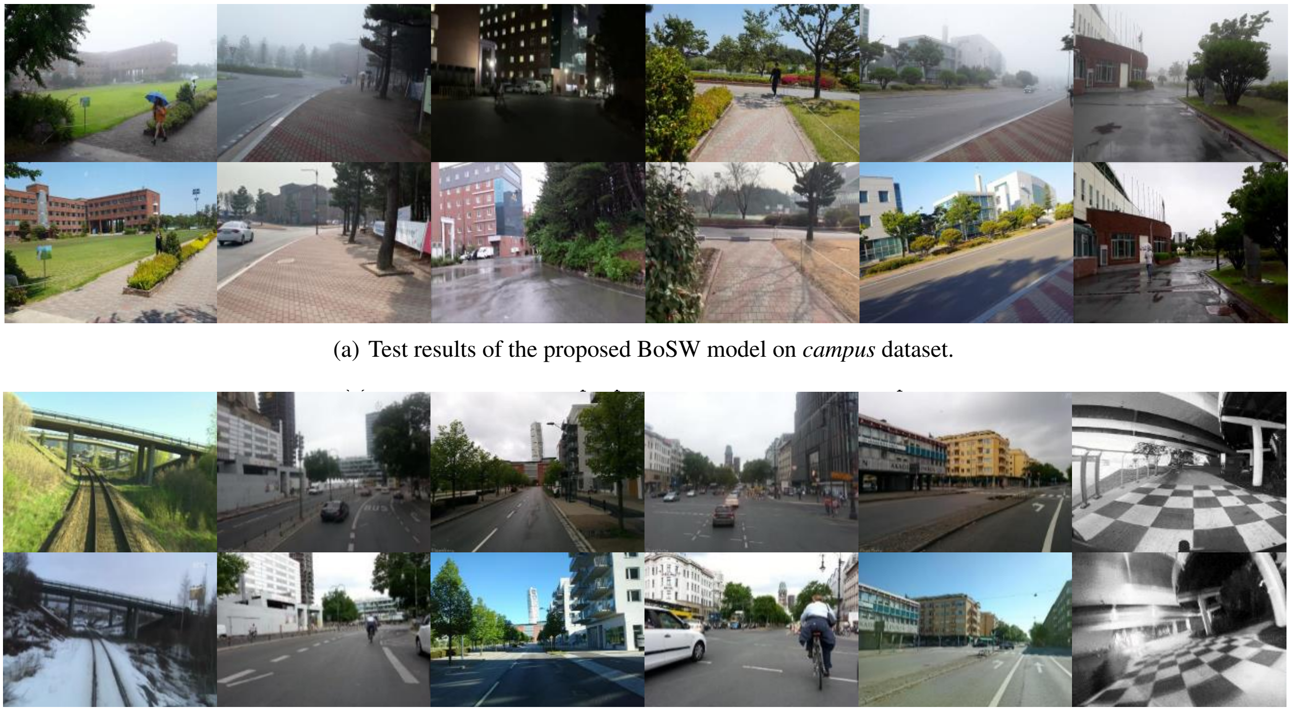
The examples of the proposed method in challenging environments changes are as shown in Fig. 6. In the comparisons in Section 4.5 and Section 4.6, we can see the proposed method outperforms the existing BoW model in terms of speed and accuracy. The proposed method also shows similar performance on VLAD-based method. The proposed method is needed to improve recall: since the proposed methods have the best precision, the localization rate will be also increased if recall is increased.

Even though the proposed BoSW reduced the computation time for place recognition, it still needs improvement for real-time operation. An alternative way to import the proposed system to Visual SLAM system is to utilize it as a time-free additional module that is used for map alignment between online and offline map in changing environments. The module detects same place where the environments are significantly changed with matching pairs, and 3D similarity between two maps is calculated by using matching pairs. Then, two maps are aligned by the 3D similarity, and the visual odometry is conducted on the aligned map. Therefore, this module functions as the module which enables localization and mapping in changing environments, and it may be operated one or just several times that is not necessary to be real-time module.

5. CONCLUSION

In this paper, we present a new strategy to improve Bag of Words (BoW) model for visual place recognition. Instead of utilizing all extracted features, the proposed strategy utilizes sampled features only for place recognition. Since processing sampled features is faster than processing the full features, it is possible to perform the geometric verification-based ranking process which is slow, yet more robust to the geometric structure of the image considering spatial information. In addition, we update the ranking list with the proposed fast and iterative refinement for the accurate detection of true positive images in ranking list. Experimental results show that the proposed BoSW model is faster and more accurate than the existing BoW models.

As future works, we will investigate how to integrate the proposed strategy to other models such as the VLAD model. We will also study how to apply the proposed model to visual SLAM as a place recognition module for map alignment between off-line and on-line maps on changing environments.



(a) Test results of the proposed BoSW model on *campus* dataset.

(b) Test results of the proposed BoSW model on *campus VPRiCE* dataset.

Fig. 6. The examples of results of the proposed BoSW method for place recognition in the challenging environments changes. The first row of each dataset is used for query images, and the second row is searched images on database.

REFERENCES

[1] W. Zhou, H. Li, and Q. Tian, "Recent advance in content-based image retrieval: a literature survey," arXiv: 1706.06064, 2017.

[2] S. Lawry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: a survey," *IEEE trans. Robotics*, vol. 32, no. 1, pp.1-19, 2016.

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

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

[5] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2007.

[6] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2, pp. 2161-2168, 2006.

[7] J. Sivic and A. Zisserman, "Video Google: a text retrieval approach to object matching in videos," *Towar. Categ. Object Recognit.*, pp. 1470, 2003.

[8] J. L. Bentley, "K-d trees for semidynamic point sets," *Proc. Sixth Annu. Symp. Comput. Geom*, pp. 187-197, 1990.

[9] C. Silpa-Anan and R. Hartley, "Localisation using an image-map," *Australasian Conf. on Robotics and Automation*, vol. 162, 2004.

[10] M. Muja and D.G. Lowe, "Scalable nearest neighbor algorithms for high dimensional data," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2227-2240, 2014.

[11] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, 2004.

[12] A. Kelman, M. Sofka, and C. V. Stewart, "Keypoint descriptors for matching across multiple image modalities and non-linear intensity variations," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1-7, 2007.

[13] X. Zhang, L. Zhang, and H.Y. Shum, "QsRank: query-sensitive hash code ranking for efficient -neighbor search," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 2058-2065, 2012.

[14] D. Mishkin, M. Perdoch, and J. Matas, "Place Recognition with WxBS Retrieval," *CVPR 2015 Work. Vis. Place Recognit. Chang. Environ*, vol. 30, 2015.

[15] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3304-3311, 2010.

[16] S. Lowry and H. Andreasson, "Lightweight, viewpoint-invariant visual place recognition in changing environ-

ments," *IEEE Robotics and Automation Letters*, vol. 3, no. 6, pp.957-964, 2018.

[17] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier, "Large-scale image retrieval with compressed fisher vectors," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 3384-3391, 2010.

[18] D. Mishkin, J. Matas, M. Perdoch, and K. Lenc, "WxBS: wide baseline stereo generalizations," arXiv, 112, 2015.

[19] D. Mishkin, J. Matas, and M. Perdoch, "MODS: Fast and robust method for two-view matching," *Computuer Vision and Image Understanding*, vol. 141, pp. 8193, 2015.

[20] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, vol. 463, ACM press, New York, 1999.

[21] P. E. Forssén and D. G. Lowe, "Shape descriptors for maximally stable extremal regions," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 1-8, 2007.

[22] K. Mikolajczyk and C. Schmid, "Scale affine invariant interest point detectors," *Ijcv.*, vol. 60, no. 1, pp. 63-86, 2004.

[23] H. Altwaijry, A. Veit, S. J. Belongie, and C. Tech, "Learning to detect and match keypoints with deep architectures.," *Bmvc.*, vol. 60, no. 2, pp. 91-110, 2016.

[24] Law of Large Numbers, https://www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/Chapter8.pdf.

[25] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "ORB-SLAM: a versatile and accurate monocular SLAM system," *IEEE Trans. Robot.*, vol. 31, no. 5, pp. 11471163, 2015.

[26] C. Tang, O. Wang, and P. Tan, "GSLAM: initialization-robust monocular visual slam via global structure-from-motion," *Proc. - Int. Conf. 3D Vision*, 3DV 2017, 2018.

[27] J. Engel, T. Schöps, and D. Cremers, "LSD-SLAM: large-scale direct monocular SLAM," *Lect. Notes Comput. Sci.*, pp. 834-849, 2014.

[28] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," *Proc. IEEE Int. Conf. Comput. Vis.*, pp. 2564-2571, 2011.

[29] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: speeded up robust features," *Lect. Notes Comput. Sci.*, pp. 404-417, 2006.

[30] N. Sünderhauf, The VPRiCE Challenge 2015-Visual Place Recognition in Changing Environments, https://roboticvision.atlassian.net/wiki/pages/viewpage.action?pageId=14188617.

[31] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM.*, vol. 24, no. 6, pp. 381-395, 1981.

[32] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1465-1479, 2006.

[33] J. Waksberg, "Sampling methods for random digit dialing," *J. Am. Stat. Assoc.*, vol. 73, no. 361, pp. 40-46, 1978.

[34] J. S. Simonoff, *Smoothing Methods in Statistics*, Science & Business Media, Berlin, Germany, 2012.

[35] V. A. Epanechnikov, "Non-parametric estimation of a multivariate probability density," *Theory Probab. Its Appl.*, vol. 14, no. 1, pp. 153-158, 1969.

[36] L. Martino and J. Míguez, "Generalized rejection sampling schemes and applications in signal processing," *Signal Processing*, vol. 90, no. 11, pp. 2981-2995, 2010.

[37] Geometric verification of matching, http://www.micc.unifi.it/delbimbo/wp-content/uploads/2011/10/slide_corso/A34%20Geometric%20verification.pdf, Accessed 15 November 2011.

[38] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2nd eds., Cambridge, England, 2003.

[39] O. Chum, J. Matas, and J. Kittler, "Locally optimized RANSAC," *Joint Pattern Recognition Symposium*, Springer, Berlin, Heidelberg, pp. 236243. 2003.

[40] D. M. Chen, G. Baatz, K. Köser, S. S. Tsai, R. Vedantham, T. Pylvänäinen, K. Roimela, X. Chen, J. Bach, M. Pollefeys, B. Girod, and R. Grzeszczuk, "City-scale landmark identification on mobile devices," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 737-744, 2011.

[41] R. Gomez-Ojeda, M. Lopez-Antequera, N. Petkov, and J. Gonzalez-Jimenez, "Training a convolutional neural network for appearance-invariant place recognition," *CVPR Work. Vis. Place Recognit. Chang. Environ.*, 2015.

[42] O. Vysotska and C. Stachniss, "Lazy sequences matching under substantial appearance changes," *Proc. of ICRA 15 WS VPRiCE. (short paper)*, 2015.

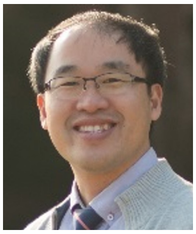
[43] M. Douze, A. Ramisa, and C. Schmid, "Combining attributes and Fisher vectors for efficient image retrieval," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 745-752, 2011.

[44] S. Zhao, H. Yao, Y. Yang, and Y. Zhang, "Affective image retrieval via multi-graph learning," *Proc. ACM Int. Conf. Multimed.*, 2014.

[45] R. Tao, A. W. M. Smeulders, and S. F. Chang, "Attributes and categories for generic instance search from one example," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 1025-1028, 2015.

[46] X. Chen and Y. Jia, "Indoor localization for mobile robots using lampshade corners as landmarks: visual system calibration, feature extraction and experiments," *International Journal of Control, Automation, and Systems*, vol. 12, no. 6, pp. 1313-1322, 2014.

Sang Jun Lee received his B.S. degree in Computer science and Engineering from Handong Global University, Pohang-si, Korea, in 2017. He is currently pursuing an M.S. degree in the Dept. of Information Technology at the Handong Global University. His research interests include the SLAM system for the localization of self-driving cars, robotics, and AR with 3D reconstruction, and optimization.



Sung Soo Hwang received his B.S. degree in Electrical Engineering and Computer Science from Handong Global Unveristy, Pohang, Korea in 2008, and his M.S. and Ph.D. degrees in Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 2010 and 2015, respectively. His research interests include image-based 3D modeling, 3D data compression, augmented reality, and Simultaneous Localization and Mapping system.