Lightweight, Viewpoint-Invariant Visual Place Recognition in Changing Environments

Stephanie Lowry and Henrik Andreasson

***Abstract*—This letter presents a viewpoint-invariant place recognition algorithm which is robust to changing environments while requiring only a small memory footprint. It demonstrates that condition-invariant local features can be combined with Vectors of Locally Aggregated Descriptors to reduce high-dimensional representations of images to compact binary signatures while retaining place matching capability across visually dissimilar conditions. This system provides a speed-up of two orders of magnitude over direct feature matching, and outperforms a bag-of-visual-words approach with near-identical computation speed and memory footprint. The experimental results show that single-image place matching from nonaligned images can be achieved in visually changing environments with as few as 256 b (32 B) per image.**

***Index Terms*—Visual-based navigation, recognition, localization.**

I. Introduction

THE goal of a visual place recognition system is to recognize whether a robot has visited its current location on a previous occasion. Ideally, a visual place recognition system should recognize a place even when the appearance of the place has changed (see Fig. 1); it should recognize a place even if it is observing the place from a different viewpoint (see Fig. 1(c) and (d)); it should be efficient at comparing its current view to all previously visited places in its database; and the memory required to store each place should be as small as possible, so many places can be remembered.

These issues can be difficult to simultaneously resolve. For example, good place recognition performance in changing environments can be achieved by systems that assume a place is always viewed from the same viewpoint. However, if this assumption is violated the system will fail [1]. Systems that are robust to both appearance and viewpoint changes typically depend on rich, high-dimensional descriptors and direct feature matching to perform effective but computationally inefficient place matching [2], while efficient and viewpoint-invariant place recognition systems often fail in changing environments [3].

Manuscript received September 8, 2017; accepted December 28, 2017. Date of publication January 15, 2018; date of current version February 1, 2018. This letter was recommended for publication by Associate Editor E. M. Mouaddib and Editor F. Chaumette upon evaluation of the reviewers' comments. This work was supported by the Semantic Robots Research Profile, funded by the Swedish Knowledge Foundation. *(Corresponding author: Stephanie Lowry.)*

The authors are with the Centre for Applied Autonomous Sensor Systems, Örebro University, Örebro 70281, Sweden (e-mail: stephanie.lowry@oru.se; henrik.andreasson@oru.se).

Digital Object Identifier 10.1109/LRA.2018.2793308

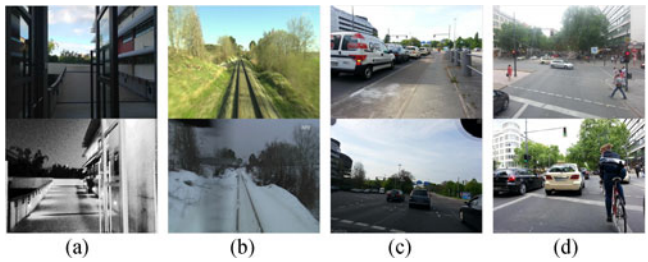

Fig. 1. Mobile robots operating in changing environments need visual place recognition systems that can cope with seasonal, weather, lighting, and illumination changes, as well as different viewpoints.

This letter investigates the potential for *condition-invariant* and *viewpoint-invariant* place recognition that can handle both viewpoint changes and appearance changes (such as those caused by changes in lighting, weather, or seasonal variation), and is also efficient and uses a small memory footprint. It combines descriptors that are robust to appearance change with feature quantization methods that reduce the amount of memory required and allow more efficient image matching.

This letter demonstrates that a system that combines condition-invariant features with an efficient and lightweight image description mechanism such as Vectors of Locally Aggregated Descriptors (VLAD) can perform place recognition even when there are both appearance and viewpoint changes. The performance of a VLAD-based system gracefully declines as the amount of memory allocated per location is reduced. VLAD has similar computational efficiency as a bag-of-words (BOW) model but achieves overall higher performance for the same memory footprint on datasets exhibiting challenging appearance change.

II. Prior Work

This section reviews relevant research on visual place recognition, with a particular focus on the place recognition goals of viewpoint-invariance, condition-invariance, and efficiency. It is common for a place recognition system to achieve two of these goals: for example, viewpoint-invariance and efficiency [4]; condition-invariance and efficiency [5]; or viewpoint-invariance and condition-invariance [3].

Viewpoint-invariance is a highly desirable characteristic for a place recognition system, as a viewpoint-invariant system can identify a location even if the camera position is

different. Viewpoint-invariant place recognition can be performed using local feature detectors such as SURF [6]. These local feature detectors identify keypoints in the image, and the image is described based on information extracted from around these keypoints.

Matching places by comparing the features in each image is a computationally intensive process. In contrast, the bag-of-visual-words (BOW) [7] model allows efficient place matching and has been used in many place recognition systems [4]. BOW quantizes the feature space into a set of k clusters, each of which represents a visual "word". Image features are allocated to a word depending on their location in the feature space. BOW allows images to be described by a binary string, and image comparison reduces to binary string matching.

The BOW model can be enhanced by many techniques that include additional information while still retaining efficient performance. Techniques include Hamming Embedding [8], Vectors of Locally Aggregated Descriptors (VLAD) [9], burstiness control [10], probabilistic image matching [4], and spatial verification methods [11]. Methods for efficiently processing large vocabularies and datasets have also been proposed [12]. More recent developments include NetVLAD [13], a neural network architecture that allows simultaneous end-of-end training of both image features and a generalized VLAD layer specifically for place recognition.

While the bag-of-words model and its derivatives provides an efficient and viewpoint-invariant method of place recognition, place recognition systems based on local features and BOW are generally not robust to appearance change [2], [3], [14]. One approach to solving the condition-invariance problem is to sacrifice viewpoint-invariance, and use whole-image features which handle appearance change well, but do not permit significant viewpoint changes. Image sequence filtering methods such as SeqSLAM [14] assume not only that the camera viewpoint is the same, but that the path through the environment is repeated. If these viewpoint-invariance and path-invariance assumptions hold, then such systems can perform condition-invariant place recognition, even using an extremely small memory footprint [5].

A compromise between point features and whole-image methods is to use image regions [2], [3], [15]. Using image regions combined with robust descriptors and matching features directly between images, these systems are computationally expensive and have high memory requirements, but can perform viewpoint-invariant and condition-invariant place recognition.

III. Approach

The goal of this letter is to combine the robust descriptors used for visual place recognition in changing environments with the low memory requirements and faster image matching provided by feature quantization techniques, such as the BOW and VLAD models. VLAD was selected for this work as it has been shown to outperform the standard BOW model for general place recognition tasks [9]. This section outlines the techniques and algorithms used.

A. Feature Detector

This work uses the SURF detector [6]. The SURF detector is less robust to appearance change than other detectors proposed for viewpoint-invariant and condition-invariant place recognition [3], [16] but provides a good compromise between robustness and efficiency. To compute the descriptors, the region of interest about each keypoint was defined as a patch of size $20s \times 20s$, where s is the detected SURF keypoint scale, as specified in [6].

B. Feature Descriptor

The selected descriptor is based on the Histogram of Oriented Gradients (HOG) [17]. HOG has been shown to be effective for place recognition in changing environments [3]. The HOG feature was selected as it provided a trade-off between efficiency and robustness, but any descriptor that demonstrates robustness to condition change – such as those used in [2], [15] – could be used.

Each image patch selected by the SURF detector was subdivided into $N \times N$ cells, and the gradient vector at each point calculated using a convolution with the horizontal and vertical filters $(1 \quad 0 \quad -1)$ and $(1 \quad 0 \quad -1)^T$ and returning the magnitude and direction of the resulting vector. Each gradient vector was added to a histogram bin of orientations divided into b bins between 0° and 180° according to the magnitude and orientation of the vector.

These features have dimensionality $d = N^2 b$. However, in this work the extracted features were reduced in dimension using Principal Component Analysis (PCA) [18] with a pre-trained PCA basis prior to use (see Section IV-B for more details).

C. Bag-of-Words Model

The bag-of-words model partitions the feature space of HOG descriptors into k visual words obtained by k-means clustering using the cosine distance. Each descriptor is assigned to the closest centroid within the feature space. An image can thus be represented by a binary string of length k where the j-th bit is 1 if and only if the j-th visual word appears in the image.

D. VLAD

Like the bag-of-words model, VLAD assigns each feature to a particular word, but while the bag-of-words includes only the binary information of whether or not the word appears in the image, VLAD extends this model by also storing position information about the feature inside the cell relative to the centroid. If multiple features can be found within the same cell, VLAD sums (or "aggregates") the relative positions together.

Specifically, the VLAD vector v is a concatenation of subvectors $v_1, v_2, \ldots, v_k$, each representing a particular visual word. For any $i \leq k$, the subvector v_i associated with the centroid c_i is defined by

$$v_i = \sum_{\substack{x \text{ such that} \\ \text{word}(x)=i}} x_j - c_i. \tag{1}$$

The vector v is L_2-normalized by $v = \frac{v}{||v||^2}$.

E. Dimension Reduction

A VLAD representation can become quite large: the size of the VLAD descriptor is $d \times k$, where d is the number of feature dimensions and k is the number of words in the bag-of-words model. For example, if a vocabulary of size 1024 and a descriptor of size 512 are used, the resulting VLAD descriptor will have 524,288 dimensions. Therefore dimensionality reduction is performed on the VLAD descriptors as a final step before storage and image matching.

The original VLAD implementation [9] reduced the dimensionality using PCA followed by Product Quantization [19]. However, data-dependent dimensionality reduction on a large VLAD descriptor requires substantial data and computation time for training, so instead we use a data-independent dimensionality reduction based on locality-sensitive hashing (LSH) that randomly projects points to a low-dimensional binary signature. This process approximately preserves the cosine similarity between the original vectors by the Hamming distance between the binary signatures [20].

We use the same random projection for each vocabulary word. This simplification reduces storage requirements but requires that the number of words k in the vocabulary must be smaller than – and a factor of – the number of bits B. The number of projection planes is $p = \frac{B}{k}$, each with the dimension d of the descriptor.

The values of the p planes P are randomly drawn from the unit normal distribution. For a VLAD descriptor v, the binary signature is calculated as

$$b = v^T P \geq 0. \tag{2}$$

The signature b has a size of $k \times p$, and as $p = \frac{B}{k}$, the total bit size of b is B.

F. Image Comparison

Image comparison is achieved by calculating a Hamming distance on binary image signatures. If a bag-of-words model containing k words is used, then the Hamming distance for two signatures b_1 and b_2 is

$$H(b_1, b_2) = \frac{\sum_{i=1}^{k} b_{1,i} \oplus b_{2,i}}{k} \tag{3}$$

where $\oplus$ is the exclusive-OR operator. For binary image signatures generated from VLAD as described Section III-E, an intra-normalization step (analogous to that proposed in [21] for L_2 normalization) is added. Thus to compare two image signatures b_1 and b_2, each consisting of k binary subvectors each of length p, we calculate the Hamming distance h_m on the m-th subvectors b_1^m, b_2^m

$$h_m(b_1, b_2) = \frac{\sum_{i=1}^{p} b_{1,i}^m \oplus b_{2,i}^m}{p}. \tag{4}$$

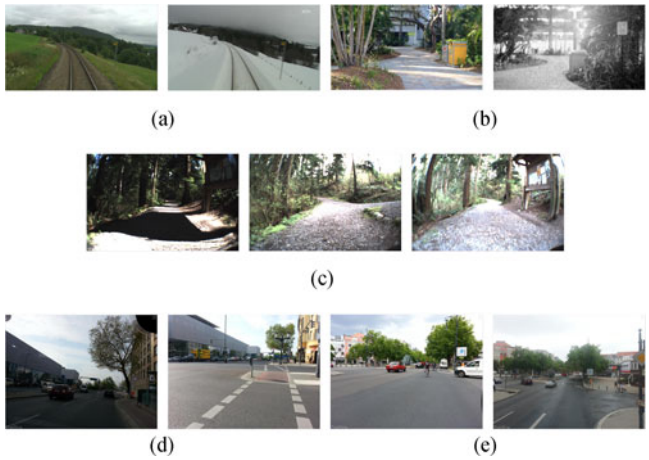

Fig. 2. The same locations but different conditions: Sample images for the (a) Nordland, (b) Gardens Point, (c) Mountain, (d) Halenseestraße, (e) Kurfürstendamm datasets.

The full Hamming distance H between b_1 and b_2 is the sum of the Hamming distances on each of the subvectors:

$$H(b_1, b_2) = \sum_{m=1}^{k} h_m. \tag{5}$$

The best matching place relative to the current location is defined as the place with the smallest binary signature distance H.

IV. Experimental Setup

This section briefly introduces the test and training datasets used to evaluate the place recognition system and summarizes key parameters and evaluation metrics.

A. Datasets

The visual place recognition system was evaluated on five publicly available datasets, each of which contains two (or more) traversals of an environment under different appearance conditions. Fig. 2 displays sample images from each dataset. The Gardens Point dataset consists of a path through a university campus in Brisbane, Australia. This path was traversed during the day and night, with lateral viewpoint changes between the paths. The daytime images were resized using bicubic interpolation to match the night-time image resolution of 640×360. The SFU Mountain dataset was captured over a 4 km forest trail in British Columbia, Canada [22]. This dataset contains 239 places observed under different weather conditions and different times of day. To make place recognition more challenging, the images of resolution 752×480 from each traversal were cropped so that different traversals were offset by 40% relative to each other. The Nordland dataset consists of images of resolution 640×360 captured from a train in Norway in different seasons. No additional offset was added to these images, which exhibited near-perfect alignment. Most of the experiments used 250 images from the Nordland dataset (starting from image 10500). The final experiment used approximately 6000 images from each traversal to test the system on a larger

TABLE I
SYSTEM PARAMETERS

SURF Detector	
Number of scales	4
Number of octaves	4
HOG Descriptor	
Number of blocks	14
Number of orientations	9
Total size	$14^2 \cdot 9 = 1764$
Full Image Descriptor	
Features extracted per image	300 Also 100, 800, 2000 in Section V-B
Feature dimension	8, 16, 32, 64, 128, 256, 512, 1024
Vocabulary size	8, 16, 32, 64, 128, 256, 1024, 4096, 16384
VLAD signature length (bits)	64, 256, 1024, 2048, 4096, 16384

dataset, and the benchmarking experiments used 29000 images. Finally, the Berlin Halenseestraße and Berlin Kurfürstendamm datasets were introduced in [2] and consist of images of resolution 640×480 sourced from the Mapillary image-sharing service.[1] The image streams are captured from different vehicles including a car, a bicycle, and the top of a bus, so places are seen from widely disparate viewpoints. There is also variation in the lighting and weather conditions within the datasets.

B. Training

The system was trained on 500 images from the Nordland summer dataset, from which 441,538 features were extracted. These training features were used to calculated the PCA basis for the initial dimension reduction on the HOG descriptors, and to generate a vocabulary model via k-means clustering for the bag-of-words.

The same PCA basis and vocabulary model were used on all the test datasets, to ensure that the system was widely generalizable and did not require special tuning for each scenario.

C. Parameters

Table I presents some key experimental parameters. The parameter values for the SURF detector and HOG descriptor remained constant through the experiments, while multiple parameter values for the feature dimension, vocabulary size and VLAD signature length were used and the range of tested values is shown here. In the majority of experiments, 300 features were extracted from each image, with the exception of Section V-B where between 100 and 2000 features were used.

D. Implementation

The system was implemented in Matlab,[2] using the built-in functions to compute the SURF keypoints, the PCA decomposition, and the HOG descriptor. Both BOW and VLAD were implemented using the Yael library [23]. Timing comparisons were performed on a commerical laptop using a single core of an Intel i7-4810MQ CPU.

[1]http:www.mapillary.com

[2]The code is available for download at https://github.com/short-circuitt/LWPR

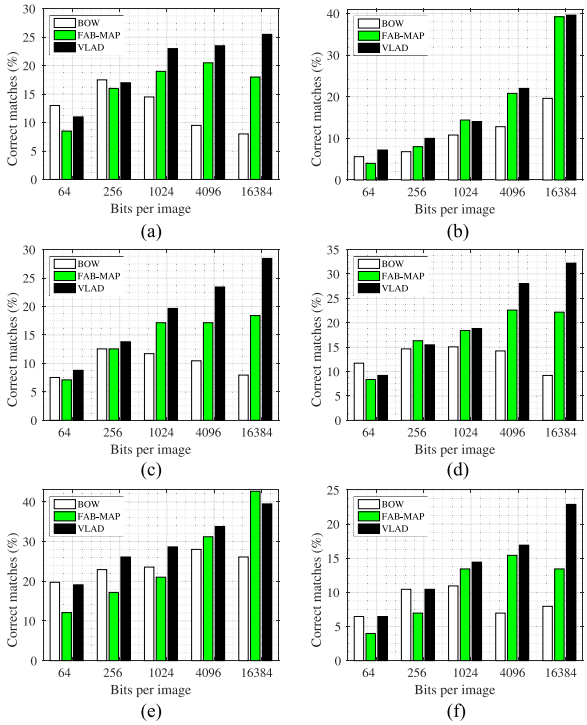

Fig. 3. Percentage of places correctly matched by BOW (white bars), FAB-MAP (green bars), and VLAD (black bars). VLAD matches a higher percentage of places correctly for most bit length and dataset combinations. (a) Gardens Point Day-Night. (b) Nordland Summer-Winter. (c) Mountain Dry-Wet. (d) Mountain Dry-Dusk. (e) Halenseestraße. (f) Kurfürstendamm.

E. Evaluation

The key metric of evaluation was the percentage of correct matches. For each image, the best match was selected according to the Hamming distance as defined in Section III-F. If the best match was sufficiently close to the correct match, within the tolerance level which was set to 2 images for each dataset, it was considered a true positive match.

V. RESULTS

The results evaluate the performance of the visual place recognition system with respect to the amount of information stored per image as well as the computation time for image processing and comparison. The VLAD-based system is also compared to other place recognition approaches; namely, BOW, SeqSLAM, FAB-MAP, and full feature matching.

A. Image Signatures

A key requirement for place recognition systems is that the stored descriptor size is as small as possible. This experiment investigated how place recognition performance related to the amount of information stored per image. Fig. 3 shows the performance of BOW, FAB-MAP (using the OpenFABMAP implementation [24]) and VLAD using the same number of bits per image.

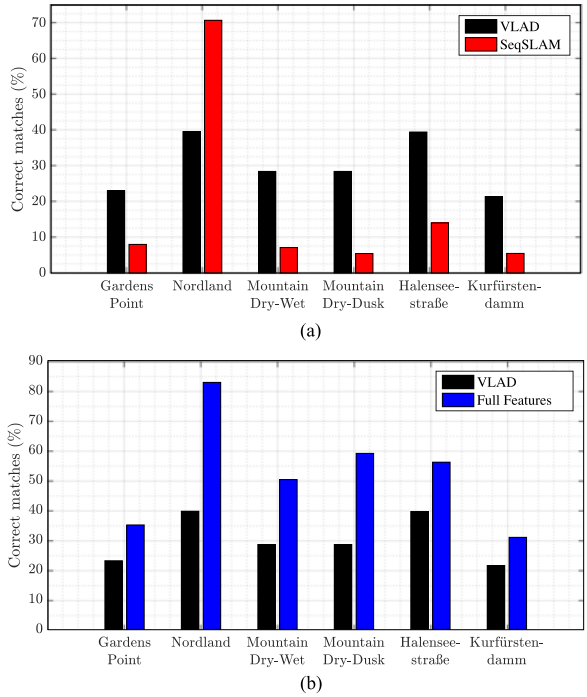

Fig. 4. (a) Correct matches achieved by VLAD compared to SeqSLAM. SeqSLAM only outperforms VLAD on the aligned track-based Nordland dataset. (b) Correct matches achieved by VLAD compared to direct feature matching. The feature matching outperforms VLAD in each case, most effectively on the Nordland dataset. However, it requires 1000 times as much memory.

VLAD outperforms both BOW and FAB-MAP for most bit length and dataset combinations. An exception is in Fig. 3(e) using 16384 bits where FAB-MAP matches 43% of places correctly compared to VLAD's 39%. However, in the other datasets VLAD outperforms FAB-MAP using 16384 bits by as much as 10% of the dataset (13% correct matches compared to 23% correct matches in Fig. 3(f)).

Interestingly, BOW outperforms both VLAD and FAB-MAP in several cases when the number of bits per image is very small (see Fig. 3(a) and (d)–(f)). However, the number of bits stored per image increases, BOW is consistently outperformed by both VLAD and FAB-MAP.

The performance of BOW does not always improve when more bits are stored per image, and in four of the six datasets worse performance is achieved with a vocabulary of 16384 bits than with 256 bits (see Fig. 3(a), (c), (d), and (f)). As noted in [25], the word clusters for a BOW model must be large enough to capture all the variation in the descriptors due to appearance and viewpoint change, but not so large that too many distinct features are clustered together. Thus an intermediate sized vocabulary provides a trade-off between these competing priorities. In contrast, the performance of VLAD is consistently related to the number of bits per image – if the VLAD signature uses more bits, VLAD will match a larger number of places correctly.

Fig. 4(a) compares the performance of VLAD using 16384 bits to SeqSLAM [14], using the OpenSeqSLAM implementation with default parameters [1]. SeqSLAM also stores 16384 bits per place, in the form of a 32×64 pixel 8-bit image. VLAD outperforms SeqSLAM on all datasets except the well-aligned Nordland dataset despite SeqSLAM using multiple images to decide place matching. SeqSLAM's poor performance on the other datasets is likely due to its assumption of viewpoint invariance, which is violated in these cases. In contrast, VLAD does not make any viewpoint assumptions.

Finally, Fig. 4(b) compares the performance of VLAD using 16384 bits to a version that stores the same features in uncoded form. Each feature is 1764 dimensions and 300 features are stored per image, so (using 4 bytes to encode each dimension) the total number of bytes stored is $1764 \times 300 \times 4$ bytes or around 2 MB of data per image. The uncoded features achieve performance 2.7 times greater than the 2048-byte VLAD descriptor on the Nordland dataset, and large improvements are also observed on the Mountain and Halenseestraße datasets. However, around 1000 places can be stored as VLAD-16384 for every full feature image, and depending on the requirements of the system an even smaller VLAD descriptor could be used if a large number of images must be stored.

TABLE II
IMAGE PROCESSING TIMES

	Mean processing time per image (s)
Feature detection and extraction (300 features)	0.786
Compute BOW (16384 words)	0.088
Compute BOW (128 words)	0.002
Compute VLAD (128 words, 16384 bits)	0.003

B. Computation Time

A place recognition system should ideally be computationally efficient as well as effective at performing location matching. The required processing can be broken down into two separate stages, the image processing stage, and the image comparison stage.

1) Image Processing: For both VLAD and BOW, as well as for full feature matching, the following steps are performed:

- Keypoints are detected in an image.
- Descriptors are extracted based on these keypoints.
- The descriptors are reduced using PCA.

For BOW, the following process must also be performed:

- Each descriptor is matched to a visual word by finding the nearest neighbor cluster center.

For VLAD, the following processes must also be performed:

- As for BOW, each descriptor is matched to a visual word using nearest neighbor matching.
- The difference between each descriptor and the cluster center is calculated and aggregated.
- The vector is L_2-normalized.
- The binary signature is calculated.

Benchmarking experiments were performed using 29,000 images from the Nordland dataset. The computation time for keypoint detection and feature extraction was separated from the later processing steps. The average processing time per image for 16384-bit BOW and VLAD are shown in Table II.

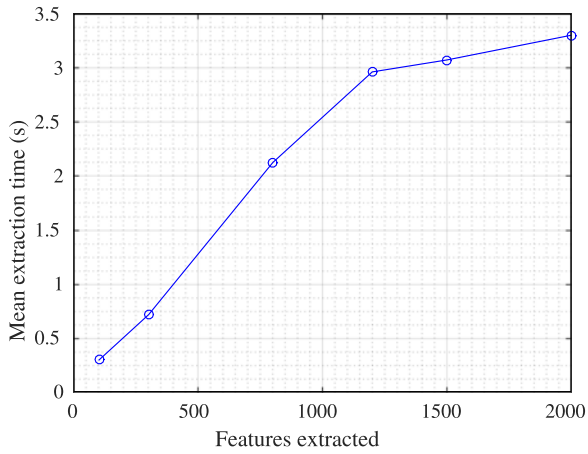

Fig. 5. Feature extraction times for the HOG descriptor used in the letter. More time is required as more features are extracted from an image.

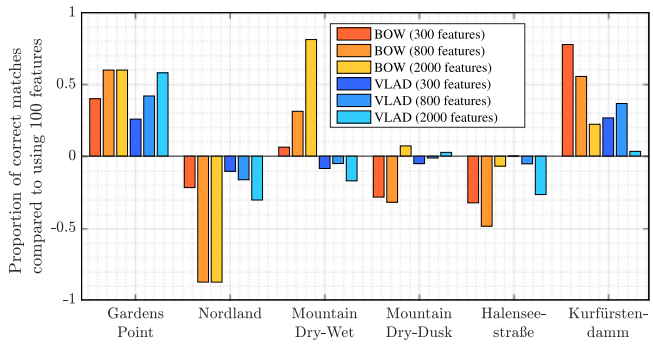

Fig. 6. BOW and VLAD performance for different numbers of features per image relative to performance using 100 features. Adding more features does not always improve the system performance for either BOW or VLAD. Compared to VLAD, BOW's performance varies more with changing feature number.

VLAD is actually more efficient than BOW for the same bit length despite requiring additional processing steps. The BOW model uses a larger vocabulary, and when features are assigned to a visual word, the nearest neighbor calculation depends on the number of clusters. In this case BOW's vocabulary size of 16384 compared to VLAD's 128 results in a slower nearest neighbor calculation, which takes a longer time than the subsequent VLAD processing.

However, the computation is dominated by the feature extraction process, and this process depends critically on the number of features extracted per image (see Fig. 5). A system that requires fewer features per image will be more efficient. Fig. 6 compares the performance of VLAD and BOW for different numbers of features per image. These results are presented relative to the system performance using 100 features; each bar represents $\frac{c_i - c_{100}}{c_{100}}$ where c_i is the percentage of correct matches when i features are used.

Overall, these results demonstrate that more features do not necessarily provide better performance for either BOW or VLAD (see the Nordland, Mountain, and Berlin Halenseestraße datasets). Furthermore, BOW demonstrates more sensitivity to the choice of parameters. The changes (either negative or positive) for BOW are higher than for VLAD (with the exception of the Gardens Point dataset). This demonstrates BOW's sensitivity to the choice of vocabulary – if the relationship between the features and the word clusters changes at all (say, by adding or removing features), the performance change can be very large. The number of features extracted has a less critical effect on VLAD.

TABLE III
IMAGE COMPARISON TIMES

	Mean time for 10,000 images (s)
BOW (16384 bits=2048 bytes)	0.84
VLAD (16384 bits=2048 bytes)	0.78
Full features (2 MB)	80.3

2) Image Comparison: Image comparison is very similar for BOW and VLAD: both systems use a version of the Hamming distance to compare images together. The mean comparison times for performing 10,000 image comparisons using Nordland dataset images is presented in Table III. There is little timing difference between the two approaches. In contrast, direct feature matching is two orders of magnitude slower.

Although BOW and VLAD have similar comparison times, BOW has an advantage as it is sparser than VLAD: a BOW signature typically contains more zeros than a VLAD signature. On the Nordland dataset, the median number of ones per BOW signature is 237, while the median number of ones for VLAD is 1216. This sparseness gives the potential for other more efficient comparison methods to be applied to BOW [26]. However, as VLAD has better place recognition performance than BOW, this additional efficiency comes at a performance cost.

C. Vocabulary Size

An important parameter for VLAD is the choice of vocabulary size. Fig. 7 shows the relationship between performance and vocabulary size, for image signatures of various bit-lengths. On all the datasets, the best performing vocabulary size depends on the signature length. Smaller vocabularies perform well when the signature length is small; for each dataset, a vocabulary of only 8 or 16 words performs best on the 64-bit signature. If a large signature is used then larger vocabularies performs better. However, even for the 16384-bit signature, the largest vocabulary (4096 words) is out-performed by smaller vocabularies of 64 or 256 words. These results suggest that in general a small vocabulary is preferable to a larger one, particularly if a very compact image signature is required.

D. Recall at 100% Precision

The results thus far have evaluated the system performance on the percentage of correct matches. Another frequently used metric for place recognition is the recall at 100% precision. Recall and precision are defined as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

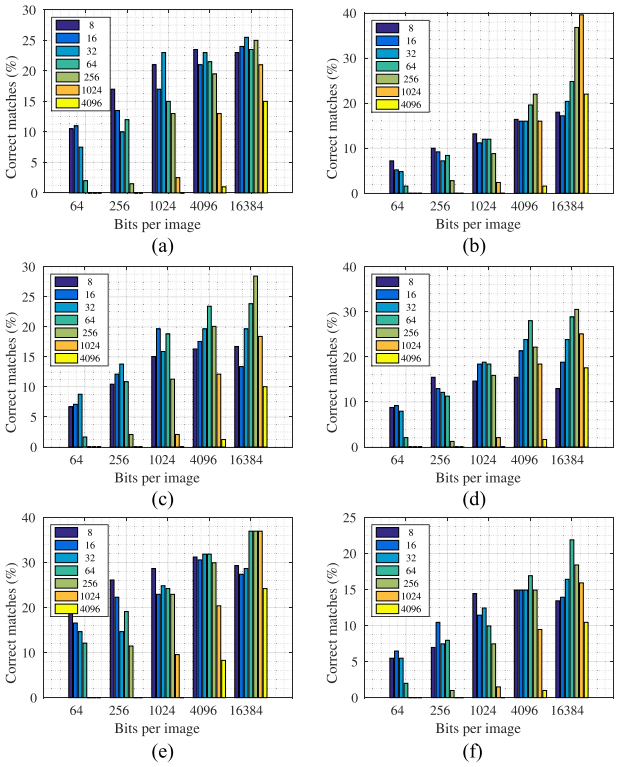

Fig. 7. Correct matches for combinations of vocabulary size and signature length. Smaller vocabularies (8, 16, or 32 words) typically perform better than large vocabularies when the image signature is short (64 or 256 bits). For a longer image signature (16384 bits) a larger vocabulary of 64 or 256 words can be used. The largest vocabulary of 4096 words was out-performed by almost all the smaller vocabularies. (a) Gardens Point Day-Night. (b) Nordland Summer-Winter. (c) Mountain Dry-Wet. (d) Mountain Dry-Dusk. (e) Halenseestraße. (f) Kurfürstendamm.

Here TP is the number of true positive matches, FP is the number of false positive matches and FN is the number of false negative matches. The recall at 100% precision calculates what proportion of true matches exceed a threshold that no false positive matches achieve. Fig. 8(a) shows the recall at 100% precision for the Nordland dataset, and for comparison displays the performance in terms of correct matches in Fig. 8(b). The recall at 100% precision is severely affected by the reduction in bit number, dropping to close to zero for signatures of 2048 bits or shorter. In contrast, the percentage of correct matches for these smaller image signatures is diminished compared to a larger signature but does not completely collapse, remaining at nearly 20% for 2048 bits.

A compromise approach to the loss of high-precision recall in a system is to perform a filtering step. This work uses an image sequencing filter based on SeqSLAM. We use the OpenSeqSLAM filtering code [1] with default parameters but using VLAD-2048 instead of image pixels as the descriptor for each location. Fig. 8(c) displays the precision-recall curve on the Nordland dataset for the 2048-bit signature. With the additional filtering step, the recall at 100% precision rises to 35%. However, Fig. 8(d) shows that for signatures that are 1024 bits or smaller, recall at 100% precision is very low even for SeqVLAD. This result demonstrates that recall at 100% precision is more sensitive to image signature length than finding correct matches, even when filtering is performed.

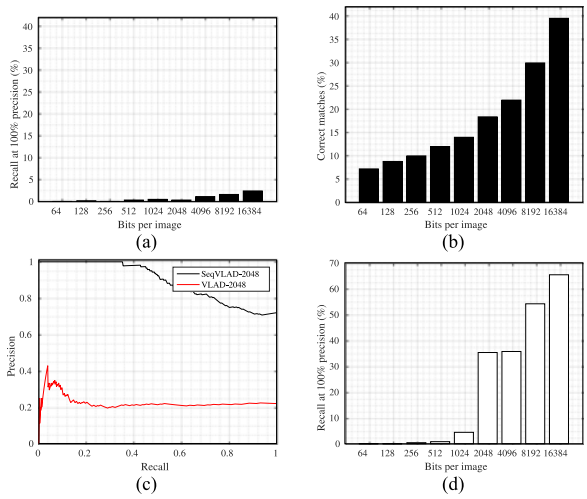

Fig. 8. VLAD performance on Nordland for different bit lengths. (a) VLAD achieves less than 3% recall at 100% precision for all signature lengths, although (b) up to 40% of matches are correct. (c) When a sequence filter SeqVLAD is added, recall at 100% precision for a 2048-bit signature is 35%. (d) However, below 2048 bits recall at 100% precision drops off steeply.

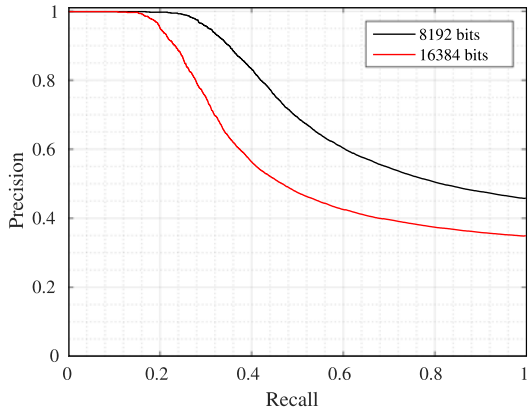

Fig. 9. Precision-recall curve for SeqVLAD on full Nordland dataset. The smaller signature (8192 bits) outperforms the larger signature (16384 bits) with a recall at 100% precision of 17%.

E. Long Dataset

Fig. 9 displays the precision-recall curve for SeqVLAD on the full Nordland dataset, excluding the training set (the first 6000 images). Images were sampled at 0.2 Hz for a total of 5954 images. Two signature lengths were tested: 8192 bits with a feature dimension of 256 and a vocabulary size of 128, and 16384 bits with a feature dimension of 512 and a vocabulary size of 1024. The shorter VLAD signature with the smaller features and vocabulary outperformed the longer signature with a larger vocabulary, with a recall at 100% precision of 17% compared to 11%. These results shows that, when combined with a filtering method, VLAD can provide high-precision place recognition with low memory requirements over a large and visually changing environment.

VI. Conclusion

This letter demonstrates that a lightweight place recognition system can perform visual place recognition from different viewpoints, even when the appearance of the environment has changed. Unlike other viewpoint-invariant and condition-invariant place recognition systems, the approach presented here is also efficient in terms of both memory and computation.

There is a clear trade-off between memory usage and performance, and the system performance degrades as the number of bits per image decreases. However, the performance degrades slowly, matching at least 10% of places correctly in all datasets when 256 bits per image are stored, and still matching 5% of places correctly in all datasets when 64 bits per image are stored. There is also a trade-off between computation time and performance: VLAD is outperformed by an exhaustive feature matching approach, but the computation time required is two orders of magnitude longer.

Both the bag-of-words model and FAB-MAP performed well compared to previous experiments in changing environments [1], [3], [14], due to the use of a condition-invariant image descriptor, and a small vocabulary. However, while small vocabularies reduce the quantization error [25] they also lose discriminative capability. The local position information that is retained by VLAD allows the system to distinguish between different features in the same word cluster, thereby improving the performance and decreasing the sensitivity to the choice of vocabulary.

The bottleneck in the current implementation of the algorithm is the feature extraction step, and future work will focus on ensuring the descriptor extraction becomes more efficient but remains robust to appearance change. VLAD is feature agnostic, and alternative detectors and descriptors can easily be integrated into the system. Detectors and descriptors can also be selected for increased robustness or increased efficiency depending on the desired application. There are also many techniques for enhancing the standard VLAD model, such as [13], [21], which could improve performance even further.

Visual place recognition systems enhance the localization reliability of mobile robots, and are often combined with GNSS systems for robust multi-sensor localization. Cameras are preferred to LiDAR sensors when cost, passive sensing, or power limitations are a consideration, and can operate in both open and structured environments. This system contributes to reducing the required memory storage to make visual place recognition a feasible option for long-term localization in a visually changing outdoor environment.

References

[1] N. Sünderhauf, P. Neubert, and P. Protzel, "Are we there yet? Challenging SeqSLAM on a 3000 km journey across all four seasons," presented at *ICRA Workshop Long-Term Autonomy*, Karlsruhe, Germany, 2013.

[2] N. Sünderhauf *et al.*, "Place recognition with ConvNet landmarks: Viewpoint-robust, condition-robust, training-free," presented at *Robot. Sci. Syst. Conf.*, Rome, Italy, Jul. 2015.

[3] C. McManus, B. Upcroft, and P. Newmann, "Scene signatures: Localised and point-less features for localisation," presented at *Robot. Sci. Syst. Conf.*, Berkeley, CA, USA, Jul. 2014.

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

[5] M. Milford, "Vision-based place recognition: How low can you go?" *Int. J. Robot. Res.*, vol. 32, no. 7, pp. 766–789, 2013.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understanding*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[7] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2003, vol. 2, pp. 1470–1477.

[8] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. 10th Eur. Conf. Comput. Vis.*, 2008, pp. 304–317.

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

[10] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1169–1176.

[11] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.

[12] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 2, pp. 2161–2168.

[13] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PP, no. 99, pp. 1–1, 2017.

[14] M. Milford and G. Wyeth, "SeqSLAM: Visual route-based navigation for sunny summer days and stormy winter nights," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 1643–1649.

[15] P. Neubert and P. Protzel, "Local region detector + CNN based landmarks for practical place recognition in changing environments," in *Proc. Eur. Conf. Mobile Robot.*, Sep. 2015, pp. 1–6.

[16] T. Krajník, P. Cristforis, M. Nitsche, K. Kusumam, and T. Duckett, "Image features and seasons revisited," in *Proc. Eur. Conf. Mobile Robot.*, Sep. 2015, pp. 1–7.

[17] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2005, vol. 1, pp. 886–893.

[18] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer, 2002.

[19] H. Jégou, M. Douze, and C. Schmid, "Product quantization for nearest neighbor search," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, Jan. 2011.

[20] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proc. 34th Annu. ACM Symp. Theory Comput.*, 2002, pp. 380–388.

[21] R. Arandjelović and A. Zisserman, "All about VLAD," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 1578–1585.

[22] J. Bruce, J. Wawerla, and R. Vaughan, "The SFU mountain dataset: Semi-structured woodland trails under changing environmental conditions," presented at IEEE Int. Conf. Robot. Autom./Workshop Vis. Place Recognit. Changing Environ., 2015.

[23] M. Douze and H. Jégou, "The Yael library," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 687–690.

[24] A. Glover, W. Maddern, M. Warren, S. Reid, M. Milford, and G. Wyeth, "OpenFABMAP: An open source toolbox for appearance-based loop closure detection," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 4730–4735.

[25] A. Ranganathan, S. Matsumoto, and D. Ilstrup, "Towards illumination invariance for visual localization," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 3791–3798.

[26] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. Boston, MA, USA: Addison-Wesley, 1999.