Online Visual Vocabularies

Yogesh Girdhar and Gregory Dudek
Center for Intelligent Machines
McGill University
Montreal, Canada
{*yogesh,dudek*}@*cim.mcgill.ca*

***Abstract*—The idea of an online visual vocabulary is proposed. In contrast to the accepted strategy of generating vocabularies offline, using the k-means clustering over all the features extracted form all the images in a dataset, an online vocabulary is dynamic and evolves iteratively over time as new observations are made. Hence, it is much more suitable for online robotic applications, such as exploration, landmark detection, and SLAM, where the future is unknown. We present two different strategies for building online vocabularies. The first strategy produces a vocabulary, which optimizes the k-centres objective of minimizing the maximum distance of a a feature from the closest vocabulary word. The second strategy produces a vocabulary by randomly sampling from the current vocabulary and the features in the current observation. We show that both the algorithms are able to produce distance matrices which have positive rank correlation with distance matrices computed using an offline k-means vocabulary. We discover that the online random vocabulary is consistently effective at approximating the behaviour of the offline k-means vocabulary, at least for the moderate sized datasets we examine.**

***Keywords*-bag of words; visual vocabulary;**

I. INTRODUCTION

This work proposes two novel techniques for maintaining a continuously evolving vocabulary of visual words, which is suitable for online applications such as landmark detection, mapping, and summarization.

One of the most fundamental problem in computer vision is of object recognition and categorization. Matching an object or a place from a database of images is a challenging task and requires modelling the visual appearance in a viewpoint and lighting invariant manner. Recently the bag of words model for describing images [1] has become popular. In this technique, first a vocabulary of visual words is generated by clustering SIFT or SURF features extracted form all the images in the database, and then each image is expressed as a histogram of frequency count over these visual words.

The problem with such an approach is that the vocabulary must be computed offline, and hence it might not be suitable for expressing images which we have not seen before. Moreover, applications such as vision based exploration, landmark detection [2], view based mapping [3], and online summarization [4], [5], require comparing an incoming image observation with previously observed images. Hence, for such applications, a huge vocabulary is required. Nister et al. [6] have proposed a technique for building vocabulary trees to handle large vocabularies, which allow large vocabularies to be used more efficiently, but still these vocabularies are not useful for online applications which require lifelong learning.

We proposes an alternative approach to generating vocabularies for online applications. A small online vocabulary which can be updated efficiently to incorporate new observations, can replace a huge vocabulary computed offline over some other dataset. We show that that an online vocabulary can give performance comparable to an offline vocabulary, even if the offline vocabulary is computed on the same dataset, which is not possible in a real-world scenarios.

We present two novel online algorithms for generating vocabularies and compare their performance with an offline vocabulary generated using the k-means algorithm. The first algorithm finds a vocabulary which minimizes the maximum distance of a an observed feature from one of the vocabulary words. The second algorithm maintains a random selection of features observed so far, while giving more weight to the features in the current observation.

We measure the goodness of the online vocabularies by comparing them with a large offline vocabulary generated on the same dataset using the k-means algorithm. Similarity between two vocabularies is measured in terms of the mean Kendall τ rank correlation between the distance matrices associated with the two vocabularies.

The primary research contributions of this work are the following:

- it proposes the idea of using a small online vocabulary, to replace a large offline vocabulary.
- it presents two different techniques for generating an online vocabulary.
- it presents a novel way to compare two given vocabularies using the Kendall τ rank correlation coefficient.
- experimental demonstration of the effectiveness of the online vocabularies using several different datasets from different environments.

II. Problem Overview

A. The Offline Problem

Before we consider the problem of computing vocabulary online, first let us re-examine the offline problem.

Let $\mathbf{Z} = \{Z_i\}$ be the set of all the descriptions, extracted from all the images in the image database. Sivic et al. [1] proposed the use of the k-means objective, which finds the vocabulary $\mathbf{S}$ such that the cost defined as:

$$\mathrm{Cost}(\mathbf{S}) = \frac{1}{|\mathbf{Z}|} \sum_i \min_j d(Z_i, S_j), \tag{1}$$

is minimized. Here $d(.)$ is the distance function between two feature descriptions. Typically the size of the vocabulary $|\mathbf{S}| = k$ is finite, and $k << |\mathbf{Z}|$. The problem of finding the optimal solution to this minimization problem is NP-hard, and the popular k-means algorithm gives us an approximation to the optimal solution.

B. The Online Problem

Our goal is to maintain a vocabulary, which is representative of the features have been observed so far. Let $\mathbf{Z}_t$ be the set of features observed at time t, and $\mathbf{S}_{t-1}$ be the current vocabulary, representative of all the features observed so far without taking into account the current observation $\mathbf{Z}_t$. We would like to update $\mathbf{S}_{t-1}$ to get $\mathbf{S}_t$, given $\mathbf{S}_{t-1}$ and $\mathbf{Z}_t$, while minimizing

$$\mathrm{Cost}(\mathbf{S}_t) = \frac{1}{|\mathbf{Z}|} \sum_i \min_j d(Z_i, S_j), \mathbf{Z} = \cup_{i=1}^{t} \mathbf{Z}_i, \tag{2}$$

such that it is comparable to the cost of the vocabulary computed over all the observations $\mathbf{Z_1} \cdots \mathbf{Z}_t$.

C. Ideal Online Vocabulary

Although one can describe the vocabulary selection problem as a clustering problem, this does not explicitly take into account the purpose of a vocabulary, which is to be able to successfully discriminate between different images. Ideally, we would like the online vocabulary to have the same behaviour as the offline vocabulary, while finding similar or dissimilar images. To formalize this, let $d_{\mathbf{S}}(I_1, I_2)$ be the distance function which compares two images I_1 and I_2, using vocabulary $\mathbf{S}$. Let $\mathbf{S}$ be the offline vocabulary computed over all the images in the dataset, and $\mathbf{S}_t$ be the online vocabulary at time t. Then for any three images I_a, I_b, I_c, taken at times $a < t$, $b < t$ and $c < t$, we would like

$$\begin{aligned} d_{\mathbf{S}_t}(I_a, I_b) &< d_{\mathbf{S}_t}(I_a, I_c) && (3) \\ &= \begin{cases} \mathrm{T} & \text{if } d_{\mathbf{S}}(I_a, I_b) < d_{\mathbf{S}}(I_a, I_c) \\ \mathrm{F} & \text{otherwise,} \end{cases} && (4) \end{aligned}$$

i.e., we only care about the relative ordering, which the distance function imposes on the images, and not the actual distances.

III. Online Extremum Vocabulary

A good online algorithm which minimizes the k-means objective defined above in Eq. 1 is not known. A cost function closely related to the k-means objective is the k-centers cost function, defined as:

$$\mathrm{C}ost(\mathbf{S}) = \max_i \min_j d(Z_i, S_j). \tag{5}$$

Minimizing this cost function is similar to finding centres of k balls of smallest (but equal) size, which cover all the points in $\mathbf{Z}$.

If the distance function obeys the triangle inequality, then not only is the k-center problem NP-hard, but Huse and Nemhauser [7] showed that α-approximation of this problem is also NP-hard for $\alpha < 2$ (i.e. for any approximation that guarantees the cost to be at worst 2 times the optimal cost).

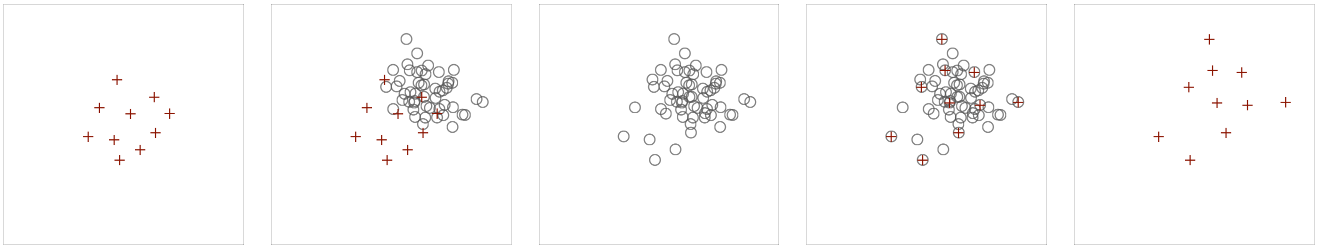
Consider the greedy strategy presented in Algorithm 1, which we refer to as the *Extremum Vocabulary* algorithm. We initialize the vocabulary with an arbitrary descriptor, then in each iteration, we choose a descriptor which is farthest away from the descriptors in the current vocabulary, and add it to the vocabulary. This algorithm has an approximation ratio of 2, and hence is likely the best we can do unless P=NP [8].

```
$\mathbf{S} \leftarrow \{Z_{\text{random}}\}$
$\mathbf{Z} \leftarrow \mathbf{Z} \setminus Z_{\text{random}}$
repeat
    $m \leftarrow \text{argmax}_i \ \min_j d(Z_i, S_j)$
    $\mathbf{S} \leftarrow \mathbf{S} \cup \{Z_m\}$
    $\mathbf{Z} \leftarrow \mathbf{Z} \setminus Z_m$
until $|\mathbf{S}| \geq k$
return $\mathbf{S}$
```

Algorithm 1: ExtremumVocabulary $(\mathbf{Z}, k)$. Computes a vocabulary as a subset of input descriptions $\mathbf{Z}$, by greedily picking the descriptions farthest away from descriptions in the current vocabulary.

Using Algorithm 1, we propose the following strategy for updating the vocabulary. We first take the union of features in the current vocabulary $\mathbf{S}_{t-1}$ and the current observation $\mathbf{Z}_t$. Lets call this set of descriptions $\mathbf{Z}' = \mathbf{Z}_t \cup \mathbf{S}_{t-1}$. Our goal is now to compute a vocabulary which is a subset of this set. We propose to run the ExtremumVocabulary algorithm on $\mathbf{Z}'$ to get the new vocabulary $\mathbf{S}_t$. Fig. 1 illustrated this graphically.

In Fig.1(a) the points represented by the red plus marker correspond to the words in the current vocabulary $\mathbf{S}_{t-1}$. The set of key-point descriptions $\mathbf{Z}_t$, observed at time t are represented by grey circles in Fig.1(b). The set $\mathbf{Z}'$, comprising of all the words form the current vocabulary and the observation is shown in Fig. 1(c). We then run the ExtremumVocabulary algorithm to select a representative subset of these words. The words are shown



(a) $\mathbf{S}_{t-1}$ (b) $\mathbf{Z}_t$ (c) $\mathbf{Z}' = \mathbf{Z}_t \cup \mathbf{S}_{t-1}$ (d) $\text{SAMPLE}(\mathbf{Z}', 10)$ (e) $\mathbf{S}_t$

Figure 1. Updating Online Vocabularies. (a) points marked with the red plus symbol correspond to the words in the current vocabulary $\mathbf{S}_{t-1}$.(b) The set of features$\mathbf{Z}_t$, observed at time t are represented by grey circles. (c) The set $\mathbf{Z}'$, comprising of all the words form the current vocabulary and the observation.(d) We then run one of the proposed sampling algorithms: extremum or random, to choose a representative subset of these features. The selected features are marked with the plus sign. (e) the selected words form the new vocabulary $\mathbf{S}_t$.

```
$\mathbf{Z}' \leftarrow \mathbf{Z}_t \cup \mathbf{S}_{t-1}$
$\mathbf{S}_t \leftarrow \text{EXTREMUMVOCABULARY}(\mathbf{Z}', k)$
**return** $\mathbf{S}_t$
```

Algorithm 2: ONLINEEXTREMUMVOCABULARY $(\mathbf{Z}_t, \mathbf{S}_{t-1}, k)$. Computes a new vocabulary at time t, given the vocabulary at previous time $\mathbf{S}_{t-1}$, and the set of features $\mathbf{Z}_t$ in the current observation.

by the red plus marker in Fig. 1(d). Finally, in Fig. 1(e), we get rid of all the descriptions not picked by the EXTREMUMVOCABULARY algorithm, and return the selected words as the updated vocabulary $\mathbf{S}_t$.

IV. ONLINE RANDOM VOCABULARY

Consider the following simple strategy for updating the vocabulary. Instead of using the extremum sampling algorithm to pick the vocabulary words from the set $\mathbf{Z}'$, we randomly sample the k features from $\mathbf{Z}'$, and use it as the updated vocabulry $\mathbf{S}_t$.

Although the algorithm seems trivial, it has the following advantages: First, it is computationally very efficient compared to Algorithm 2, since we do not need to do farthest neighbour computations. Second, It gives more weight to the current observation. This is useful because at time t, we are normally only interested in comparing the observation made at time t, with the previous observations.

```
$\mathbf{Z}' \leftarrow \mathbf{Z}_t \cup \mathbf{S}_{t-1}$
$\mathbf{S}_t \leftarrow \text{RANDOMSAMPLE}(\mathbf{Z}', k)$
**return** $\mathbf{S}_t$
```

Algorithm 3: ONLINERANDOMVOCABULARY $(\mathbf{Z}_t, \mathbf{S}_{t-1}, k)$. Computes a new vocabulary at time t, given the vocabulary at previous time $\mathbf{S}_{t-1}$, and the features $\mathbf{Z}_t$ in the current observation.

V. EXPERIMENTS

To test the performance of of the proposed online vocabulary generation algorithm, we compare it to vocabulary generated offline by the k-means algorithm. Since online vocabularies can be used for many different applications, instead of focusing on a specific application, we focus our efforts on measuring how well can an online vocabulary differentiate between different images, compared to a large offline k-means vocabulary. Any results hence observed are applicable to tasks such as landmark detection, loop closing, or summary generation.

A. *Computing Distance Matrix*

We experimented with six different datasets consisting of different environments; indoors, aerial view, and under water. For each of these datasets we first computed a vocabulary of 10,000 words by extracting SURF [9] features from each image, and then running the k-means clustering algorithm. Each image in the dataset was described using a histogram of frequency counts, by matching the extracted features with the closest word in the vocabulary.

In text retrieval, it is common for one to use *tf-idf*(term frequency - inverse document frequency) to represent documents [10]. Inverse document frequency is the weight given to a word according to how common it is amongst all the documents. If a visual word is present in all the images, then it is given a weight of 0. The aggregate *idf* is difficult to compute online, hence, in this work we give every word equal weight. To compute distance between two histograms, we use symmetric KL divergence.

The distance matrix generated by comparing each image in the dataset with every other, using the k-means vocabulary, was then used to measure the goodness of the distance matrix generated using the online vocabulary.

To generate the online distance matrices, we initialized the vocabulary with the features extracted from the first image, and then for each new image, we ran Algorithm 2 to get the updated vocabulary. We then compare the image at time t with all the images observed before this time. Hence, the ith row of this lower-triangular distance matrix was computed with vocabulary generated with the ith observed image.

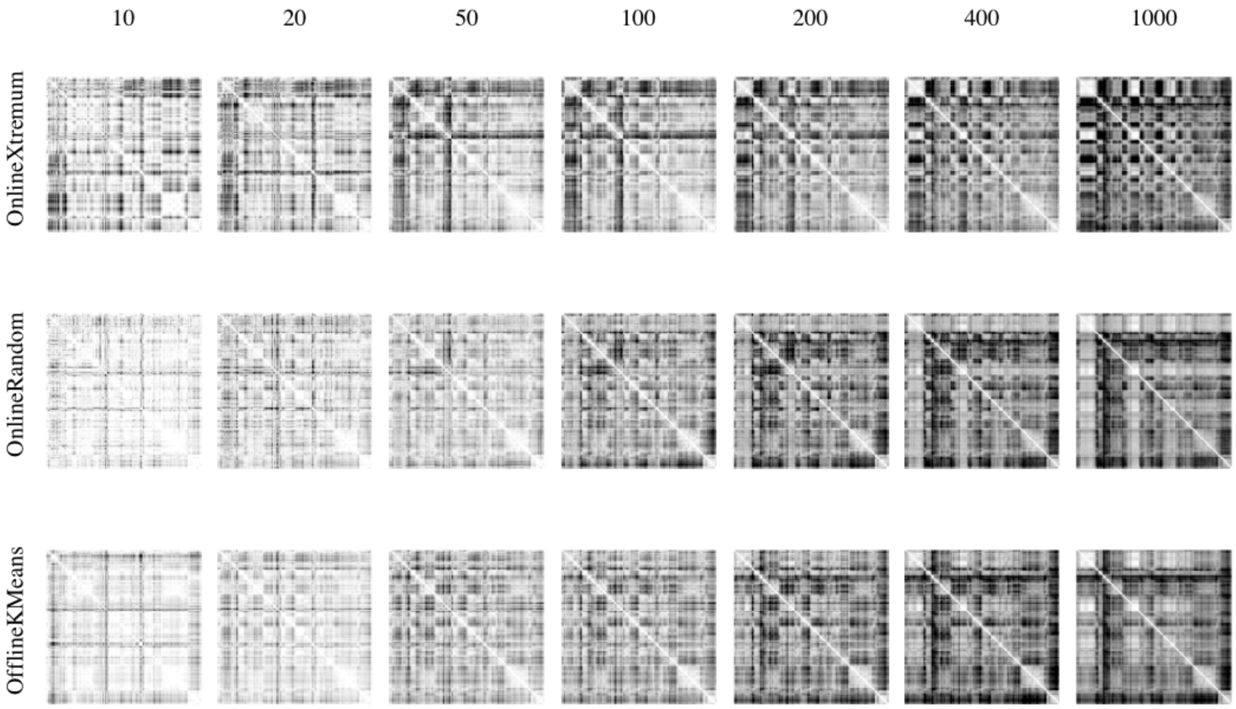


Figure 2. An example of distance matrices generated by the proposed online algorithms for different vocabulary sizes. For comparison, the distance matrix generated using offline k-means vocabulary is shown in the last row. An ideal online vocabulary should produce a distance matrix similar to the distance matrix generated using the offline k-means vocabulary.

B. Kendall τ Rank Correlation for Comparing Distance Matrices

To compare the distance matrices generated using online and offline vocabulary, we propose the use of Kendall τ Rank Correlation Coefficient [11], [12]. Kendall's τ coefficient between two lists of random variables is a measure of association based on the relative order of the consecutive elements in the two lists.

Let $X = \{x_i\}$ and $Y = \{y_i\}$ be the two lists, then the Kendall's rank correlation coefficient

$$\tau(X, Y) = \frac{n_c - n_d}{\sqrt{(n_c + n_d + n_x) + (n_c + n_d + n_y)}}, \qquad (6)$$

where n_c is the number of concordant pairs of observations, n_d is the number of discordant pairs, n_x is the number of ties involving only the elements in X, and n_y the number of ties involving only the elements in Y. A pair of observations (x_i, y_i) and (x_j, y_j) is defined as concordant if $x_i < y_i$ and $x_j < y_j$, or, if $x_i > y_i$ and $x_j > y_j$. A discordant pair is one such that $x_i < y_i$ and $x_j > y_j$, or, if $x_i > y_i$ and $x_j < y_j$.

If the elements in two lists are in exactly the same relative order, then $\tau = 1$. if the two lists are in complete disagreement (i.e., the ordering of one is reverse of the other), then $\tau = -1$. A value of $\tau = 0$ implies that the two lists are independent.

To compare two distance matrices, we compute the correlation coefficient for elements in each row i of the two matrices, and then compute the mean correlation $\overline{\tau}$. Let $d(i)$ be the ith row of of a distance matrix d. Then

$$\overline{\tau}(d_1, d_2) = \frac{1}{n} \sum \tau(d_1(i), d_2(i)), \qquad (7)$$

where n is the total number of images in the dataset.

We would like $\overline{\tau}$ computed between the online distance matrix and offline distance matrix to be > 0, and close to 1, implying that the online distance matrix will give similar classification results compared to the offline distance matrix.

VI. Results

To measure the performance of the two proposed algorithms, we first build distance matrices for vocabulary generated using each algorithm for different vocabulary sizes, and then compare them to the distance matrix generated using the vocabulary generated offline using k-means. We present results of experimentation with six different datasets each with about 100-200 images, taken from different environments.

Figure 2 shows an example of the distance matrices computed using the two proposed online algorithm for the street view dataset. The first two rows show distance matrices generated using the two proposed algorithms, and

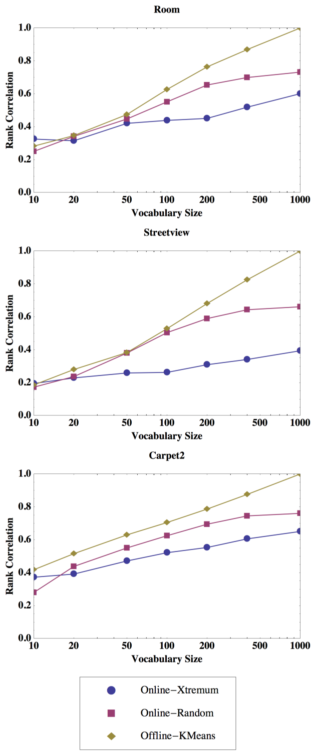


Figure 3. Correlation of distance matrices generated using the online vocabulary with the distance matrix generated using the offline k-means vocabulary of size 1000. We see that both the algorithms produce distance matrices which have positive mean rank correlation $(\overline{\tau} > 0)$ with the offline distance matrix. The correlation increases monotonically with the increase in vocabulary size. The plot also shows the correlation of the offline k-means vocabulary matrices of different sizes with the distance matrix produced using k-means vocabulary of size 1000 (shown using yellow like with diamond markers). This loosely represents the upper bound on the correlation score of the online algrotihms. See text for more details.

the third row shows the "ideal" case, i.e., the distance matrix generated using the offline k-means vocabulary. Different columns corresponds to a change in the size of the vocabulary.

The distance matrices are depicted as grayscale images, where the pixel value at row i and column j is proportional to the distance between image i and image j. White corresponds to maximum similarity and black corresponds to maximum difference. Along the diagonal we have the similarity of each image to itself, which is maximal. Typically near off-diagonal elements are also "white" since images often are correlated in time.

Intuitively, we would like the distance matrices generated by the online vocabularies to look similar to the distance matrices generated by the offline k-means vocabularies. In the example shown, it is easy to see that distance matrix corresponding to the *OnlineRandom* vocabulary is more similar to the *OfflineKMeans* vocabulary distance matrix, compared to the *OnlineXtremum* vocabulary distance matrix. The difference is most clear when we look at the last column,

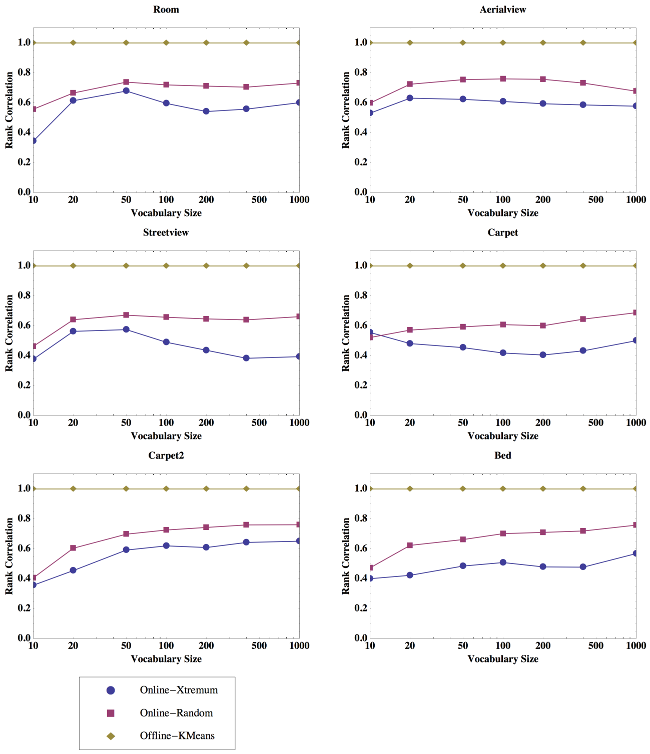


Figure 4. Correlation of distance matrices generated using the online vocabulary with the distance matrix generated using the offline k-means vocabulary of the same size as the online vocabulary.The plot corresponding to to k-means vocabulary is always 1 by definition. We see that both the algorithms are able to maintain their rank correlation score irrespective of the vocabulary size, when compared with similarly sized offline k-means vocabulary.

corresponding to the vocabulary size of 1000.

To quantify this difference, we look at two different ways of measuring the goodness of a distance matrix generated by the an online vocabulary.

First, we compare every distance matrix generated using the online vocabularies of different sizes, with the distance matrix generated by the k-means vocabulary of size 1000, using the $\overline{\tau}$ coefficient presented in Eq. 7. The results are shown in Fig. 3. We see that both the online algorithms produce distance matrices which have positive mean rank correlation $(\overline{\tau} > 0)$ with the offline distance matrix. The correlation increases monotonically with the increase in vocabulary size. Fig. 3 also shows the plot of correlation of the offline k-means vocabulary matrices of different sizes with the distance matrix produced using k-means vocabulary of size 1000. This plot (shown using yellow line with diamond markers) loosely represents the upper bound on how good an online vocabulary could be. It might seem a bit surprising that the online random vocabulary consistently outperforms the online extremum vocabulary, for all six datasets. Our hypothesis is that this is because the ExtremumVocabulary algorithm picks words which are invariant to how close they are to the dense cluster centers represented by the k-means vocabulary. The RandomVocabulary on the other hand is probabilistically more likely to pick words which are close to the cluster centres, and hence performs similar to the offline k-means vocabulary. Moreover, if one visualizes the vocabulary words as basis vectors for the set of all the words, it is known that random vectors perform quite well as basis vectors in high dimensional spaces [13].

Second, we compare the distance matrices of different sizes, with the distance matrix generated by the k-means vocabulary of the same size, as shown in Fig. 4. The plot corresponding to to k-means vocabulary is hence always 1 by definition. We see that both the algorithms are able to maintain their rank correlation score, irrespective of the vocabulary size, when compared with similarly sized offline k-means vocabulary.

VII. Conclusion

For image matching and classification, the Bag of Visual Words algorithm is powerful technique to represent image content. This technique finds use in many different computer vision applications such as classification, object categorization, view based maps, and landmark identification and, particularly, our work on image selection. The effectiveness of this representation depends, however, on the specific vocabulary of words used to build the histograms. The accepted strategy is to use the k-means clustering algorithm to cluster all the observed features, and then use the cluster centres as the vocabulary. Doing this well is only possible for offline applications, where all the data is available *a priori*.

This paper introduces the idea of *online vocabularies*, and presents two different strategies for building them. The first strategy produces vocabularies that minimize the k- centres objective function, and the second strategy produces a vocabulary by randomly sampling from the current vocabulary and the features in the last observed image.

We show that both the approaches are able to produce good results, as measured by the rank correlation between the distance matrices produced using the proposed strategy and the "ideal case", i.e., the distance matrix computed using an offline k-means vocabulary. We discover that the random summary is consistently very effective at approximating the offline k-means vocabulary, at least for the moderate sized datasets we examine.

Acknowledgment

We would like to thank Google Inc. for generously allowing us to use their street view data.

References

[1] J. Sivic and A. Zisserman, "Video google: Efficient visual search of videos," in *Toward Category-Level Object Recognition*, ser. Lecture Notes in Computer Science, J. Ponce, M. Hebert, C. Schmid, and A. Zisserman, Eds. Springer Berlin / Heidelberg, 2006, vol. 4170, pp. 127–144. [Online]. Available: {http://dx.doi.org/10.1007/11957959_7}

[2] A. Ranganathan and F. Dellaert, "Bayesian surprise and landmark detection," in *Proceedings of the 2009 IEEE international conference on Robotics and Automation*. Institute of Electrical and Electronics Engineers Inc., The, 2009, pp. 1240–1246.

[3] K. Konolige, J. Bowman, J. Chen, P. Mihelich, M. Calonder, V. Lepetit, and P. Fua, "View-based maps," *RSS'09*, 2009.

[4] Y. Girdhar and G. Dudek, "ONSUM: A system for generating online navigation summaries," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, Oct 2010, pp. 746–751.

[5] ——, "Online navigation summaries," in *IEEE International Conference on Robotics and Automation (ICRA)*, May 2010, pp. 5035–5040.

[6] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 2, 2006, pp. 2161 – 2168.

[7] W.-L. Hsu and G. L. Nemhauser, "Easy and hard bottleneck location problems," *Discrete Applied Mathematics*, vol. 1, no. 3, pp. 209 – 215, 1979. [Online]. Available: http://www.sciencedirect.com/science/article/B6TYW-46HW0N4-5/2/1a9ef82be9ca95988c9d75e76c216024

[8] T. F. Gonzalez, "Clustering to minimize the maximum intercluster distance," *Theoretical Computer Science*, vol. 38, pp. 293 – 306, 1985. [Online]. Available: http://www.sciencedirect.com/science/article/B6V1G-4968W64-P/2/3af0ff05858b1e4da7f40b02313ee908

[9] L. V. G. Herbert Bay, Tinne Tuytelaars, "Surf: Speeded up robust features," *ECCV*, 2006.

[10] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*. ACM Press / Addison-Wesley, 1999.

[11] M. G. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, no. 1-2, pp. 81–93, 1938. [Online]. Available: http://biomet.oxfordjournals.org/content/30/1-2/81.short

[12] M. Kendall, *Rank Correlation Methods*. Hafner Publishing Co., NY, 1955.

[13] D. Fradkin and D. Madigan, "Experiments with random projections for machine learning," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '03. New York, NY, USA: ACM, 2003, pp. 517–522. [Online]. Available: http://doi.acm.org/10.1145/956750.956812