Efficient clustering and quantisation of SIFT features: Exploiting characteristics of the SIFT descriptor and interest region detectors under image inversion

Jonathon S. Hare
jsh2@ecs.soton.ac.uk

Sina Samangooei
ss@ecs.soton.ac.uk

Paul H. Lewis
phl@ecs.soton.ac.uk

Electronics and Computer Science, University of Southampton
Southampton, SO17 1BJ, United Kingdom

ABSTRACT

The SIFT keypoint descriptor is a powerful approach to encoding local image description using edge orientation histograms. Through codebook construction via k-means clustering and quantisation of SIFT features we can achieve image retrieval treating images as bags-of-words. Intensity inversion of images results in distinct SIFT features for a single local image patch across the two images. Intensity inversions notwithstanding these two patches are structurally identical. Through careful reordering of the SIFT feature vectors, we can construct the SIFT feature that would have been generated from a non-inverted image patch starting with those extracted from an inverted image patch. Furthermore, through examination of the local feature detection stage, we can estimate whether a given SIFT feature belongs in the space of inverted features, or non-inverted features. Therefore we can consistently separate the space of SIFT features into two distinct subspaces. With this knowledge, we can demonstrate reduced time complexity of codebook construction via clustering by up to a factor of four and also reduce the memory consumption of the clustering algorithms while producing equivalent retrieval results.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval; H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Experimentation, Measurement, Performance, Algorithms

Keywords

Evaluation, Visual-terms, Visual-words, Image Content Analysis

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
ICMR '11, April 17-20, Trento, Italy
Copyright ©2011 ACM 978-1-4503-0336-1/11/04 ...$10.00.

1. INTRODUCTION

One of the biggest advancements in the computer vision and multimedia analysis fields over the last eight or so years has been the adoption of visual-bag-of-words representations based on the quantised SIFT descriptor. Quantised SIFT "visual term" representations are at the core of many current state-of-the-art techniques for tasks including automatic image annotation [see e.g. 2], object recognition [see e.g. 15, 4], image search [see e.g. 3] and near-duplicate detection [see e.g. 16].

The popularity of the SIFT descriptor for describing local regions is due to its robustness and invariance to small shifts in the position of the sampling region [10]. The descriptor itself is a three-dimensional histogram of gradient location and orientation. Lowe suggested that, at a given location in image scale space, gradient location can be quantised into a 4×4 location grid, and gradient angle can be quantised into 8 orientation bins [6] in order to produce a descriptor with 128 dimensions.

Sivic and Zisserman [14] originally demonstrated how SIFT descriptors could be quantised into visual words. In their approach the k-means clustering algorithm [8] was used to find clusters of SIFT descriptors. The centroids of these clusters then became the "visual words" representing the chosen vocabulary. A vector quantiser then worked by assigning local descriptors to the closest cluster. In the areas of near-duplicate detection and image search it has been shown that the size of the visual vocabulary often needs to be very large in order to achieve good performance [11]; in fact vocabularies of up to 10 million terms have been created. The biggest problem of the k-means based approach to building vocabularies is that it is computationally very expensive to create large numbers (of the order of hundreds of thousands to millions) of clusters in high (i.e. 128) dimensional spaces from massive samples of features (of the order of tens of millions). It should be noted that with such datasets, it is not only the time-complexity of the clustering algorithm that comes into play, but the inability to hold all the data being processed in memory.

Recently, two approaches have been proposed to help make the clustering of multiple SIFT features into large vocabularies more computationally tractable. Firstly, Nistér and Stewénius [11] proposed the use of hierarchical k-means to enable them to build visual vocabularies with over 1 million SIFT-based terms. The use of hierarchical k-means (HKM), also enables the vector-quantisation stage to be performed much more efficiently as it rapidly prunes the number of terms a feature must be compared against. Unfortunately the HKM algorithm has been shown to produce deficient clusters compared to normal k-means due to the way in which it partitions the space, which in turn leads to sub-optimal vocabularies. Secondly, Philbin et al. [13] demonstrated the utility of an *approximate* k-means algorithm (AKM) to achieve a cluster quality much nearer to exact k-means, but with a time complexity equivalent to the hierachical k-means technique. The approximate k-means technique works by replacing the expensive nearest-neighbour calculations required by k-means with a lookup based upon an ensemble of best-bin-first (BBF) *kd*-trees [6].

This paper shows how both the space and time requirements of k-means clustering of SIFT features for visual term vocabulary construction can be dramatically improved by directly exploiting characteristics of both the interest point detector and the SIFT feature itself.

2. DETECTORS AND DESCRIPTORS FOR LOCAL IMAGE FEATURES

In this section we highlight a few approaches to feature detection and the SIFT descriptor. The details of these techniques are explored further in the following section with regard to image inversion.

2.1 The SIFT descriptor

The SIFT descriptor is an encoding of the edge directions in a local neighbourhood producing a keypoint, given its location in an image at a scale. This encoding is designed to allow the robust comparison of identical or similar regions between images, showing resilience to additive noise, occlusion, changes in scale and orientation as well as small affine shifts [6, 1]. Given a keypoint location, detected in an image at a particular scale using one of a variety of techniques described in Section 2.2, a scale weighted window around the keypoint is used to identify the primary orientation of the edges in the region. Once identified, the primary orientation is used to align a larger scale weighted window about the keypoint location. The window itself is separated into a number of sequential bins and each edge in the window is assigned to a bin. The order of the bins is directly dependant on the primary orientation. For each edge, it's relative orientation to the primary orientation is assigned to a bin's histogram, weighted by a function of the edge's radial distance to the original keypoint. The edge's orientation is also assigned in smaller proportions to neighbouring bin histograms to allow for edge effects. The value of the components of each bin histogram make up the SIFT descriptor. Lowe recommends a 4×4 binning scheme, each bin containing 8 quantised edge directions. This results in the 128 dimensional classic SIFT descriptor.

2.2 Interest region detection

A variety of approaches can be used to locate suitable keypoints to be described by SIFT in a given image. These detection schemes employ various techniques to assure that the detected keypoints are likely to be stable under a set of transforms and additive noise. A technique, originally suggested by Lowe [6], for this is the difference-of-gaussian (DoG) approach. This takes gaussian blurs of the image at two sigma's and uses their difference image as the target of a simple neighbourhood edge detector. This edge detector is used to find stable keypoints by firstly locating points in the image which represent a local extremum of edge information, and secondly finding strong corners by calculating an approximate of the ratio of eigenvalues of a 2×2 Hessian. Importantly, given the inherent detection of extrema, the DoG is well suited to describing a given point as being either a minimum or a maximum and therefore provides the information required to treat these distinct features separately.

The Maximally stable extrema regions (MSER) detector [9] is a region detector which attempts to find stable regions. These regions can also be described using a SIFT descriptor. This detector sets an intensity threshold above which all pixels are considered not to be valid regions. Valid regions are grouped into connected regions and their size is measured. This threshold is iteratively increased, and the size of detected connected regions from the previous iteration are monitored. Those regions which maintain a size within a given threshold over a given number of iterations are considered to be stable. This approach finds only darker stable regions so the MSER process also includes an inversion step where the image is intensity inverted and the regions are found again, this time locating maxima rather than minima. This step allows the detection of a given region as being a minimum or maximum region again allowing the regions to be treated separately.

This section has discussed two feature detectors and how the features they output can be distinguished as being maxima or minima. Other feature detectors will also allow the extraction of this information prior to the description of the localised keypoints.

3. INTENSITY INVERSION AND ITS EFFECT ON THE SIFT DESCRIPTOR AND DETECTOR

Consider an image for which a set of keypoints is to be extracted. The SIFT descriptor can be used to describe local areas of an image using edge gradient information. If this image is inverted, the result is that all edge gradients are flipped. This doesn't affect the detection of keypoints, but it does result in distinct SIFT descriptors for these keypoints which are of the same visual structure with inverted intensity. This property also highlights the potential for two features in non-inverted images to be seen as distinct due to local intensity inversion rather than distinct visual structure. In this section we outline the nature of this difference and how the difference can be accounted for by applying a simple transform to the ordering of the SIFT descriptor components. Using this information, we explore how a keypoint being a local minimum or local maximum can be used to gain time-complexity optimisations in the codebook generation.

3.1 Descriptors of local minima and maxima

Consider the images in Figure 1. It is visually clear that these images are the same, with the second differing only due to an inversion of intensity such that

$$I'_{x,y} = 255 - I_{x,y} \tag{1}$$

where $I_{x,y}$ is a pixel intensity value at x, y. In Figure 2 we see a single keypoint localised by the *DoG* feature detector successfully both on the inverted and non-inverted images

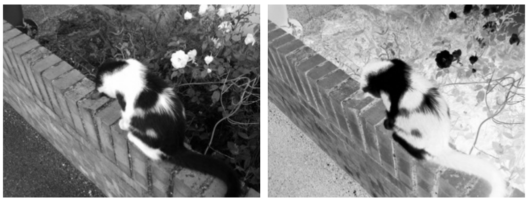

Figure 1: Starting image and inverted image generated as per Equation 1

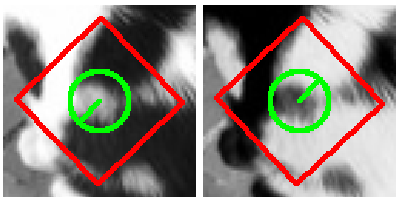

Figure 2: Two SIFT descriptor visualisations showing primary orientation, orientation window (the circle) and descriptor window (the square) for the same keypoint localised using a DoG operator. The descriptor on the left is for the keypoint in the original image, the descriptor on the right is for the keypoint in the inverted image.

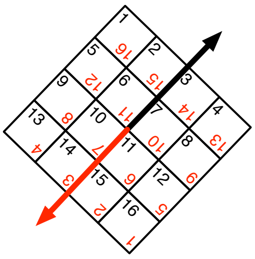

Figure 3: The ordering of SIFT descriptor bins is relative to their primary orientation. Here two descriptors with opposite primary directions are shown.

Table 1: Number of minimum and maximum (or normal/inverted) interest regions for different datasets and detectors

	DoG		MSER	
Dataset	**Min.**	**Max.**	**Normal**	**Inv.**
UKBench	7878736	7112612	9073252	14114933
MIRFlickr	10519433	10083397	31516280	39857443

along with the detected primary orientation. The primary orientation of these structurally identical points are exactly rotated 180°. In this case, one feature was a local maximum with the highest Difference-of-Gaussians in it's local neighbourhood, and another was detected as a local minimum and was the lowest local Difference-of-Gaussians. More generally, this is an example of a the different edge orientations generated when the relative intensities of neighbouring regions are flipped, resulting in flipped edge gradients. In this state, two radically different feature vectors are generated.

However, as has been explored for both mirror and inversion effects by [7], this flipping of edge orientations caused by inversion has a calculable effect on the ordering of the final SIFT descriptor. The effect of primary orientation being flipped can be countered if the order of bins is directly reversed; this is clear when studying Figure 3; by swapping bins 1 and 16, bins 2 and 15 and so on we can allow for the difference between a point found at a minimum and a point found at a maximum. Although the bin orderings are reversed to account for image inversion, the order of elements within each bin must be maintained. These elements represent a binned histogram of differences between each edge's orientation and the overall keypoint's primary orientation. Although due to image intensity inversion the edge directions have all been flipped, the primary orientation has also been flipped. This results in an identical distribution of edge orientation and primary orientation *differences* and therefore an identical relative orientation histogram between bins of the image and inverted image feature vector pair.

In summary, SIFT features detected at minima are extremely likely to have a different form to those features detected at a maxima. In the remainder of this paper we show how this difference can be exploited.

4. OPTIMISED VOCABULARY CREATION

The effect of image inversion on the SIFT descriptor suggests that the space occupied by SIFT features is rather special and with respect to inversion, could be considered to be *bimodal* and perhaps even symmetric. Because of the duality between image inversion and minima/maxima of the DoG detector (or normal/inverted MSER regions) it is possible to determine which mode of the feature space a SIFT feature will lie in at interest region detection time. These facts in turn suggest that current approaches to clustering SIFT features to create visual vocabularies are doing more work than is actually necessary and could be improved.

In the following analysis, for simplicity, we assume that the number of minimum features and number of maximum features in a dataset are equal. In reality, the actual number of features in each subset will depend on both the interest point detector and the actual image in question. Table 1 shows the number of minima and maxima detected by the difference-of-Gaussian and MSER detectors for two different image collections. In the two datasets considered, difference-of-Gaussian regions are biased slightly towards minima, and their are more inverted MSER regions than normal regions. It is fair, however, to say that the orders of magnitudes of the numbers of minimum to maximum and normal to inverted regions are of the same orders of magnitude.

The time complexity of a single iteration of standard k-means is $O(NK)$, and for hierarchical and approximate k-means this reduces to $O(N\log(K))$ where N is the number of items being clustered, and K is the number of clusters ($K << N$). Firstly, by clustering the minima and maxima separately, we can demonstrate that performance equivalent to clustering all points as one set can be achieved for a given overall vocabulary size. The time complexity of doing two clusterings is potentially dramatically reduced, because in addition to only having to cluster half the amount of data, only half the number of clusters need be produced in order to maintain the vocabulary size that would be produced from a single clustering. Defining $K_2 = \frac{K}{2}$ and $N_{max} + N_{min} = N$, then:

For standard k-means:

$$\begin{aligned} O(N_{max}K_2) + O(N_{min}K_2) &= O(K_2(N_{max} + N_{min})) \\ &= O(NK)/2 \end{aligned}$$

For HKM/AKM:

$$\begin{aligned} O(N_{max}\log(K_2)) + &O(N_{min}\log(K_2)) \\ &= O((N_{max} + N_{min})\log(K_2)) \\ &= O(N\log(K_2)) \end{aligned}$$

This rough analysis shows that just by clustering the minima and maxima features separately, but still clustering all the features to achieve the same overall vocabulary size can lead to a two-times speed-up for the standard k-means approach. Whilst the reduction in time complexity for the HKM and AKM clustering techniques is not nearly reduced as much as with standard k-means, in practice the AKM and HKM approaches still benefit massively because of the reduced memory requirements that result from having to hold half as many cluster centroids, and process half as much data at a given time.

4.1 Symmetry

If we assume that the space occupied by the SIFT features is not just two-sided, but is indeed symmetric, then a further gain can be made in performance because instead of performing two separate clusterings, we only need to perform a single clustering of the minimum (or maximum) features with K_2 clusters. From this single clustering we can then artificially invert the cluster centroids by swapping the components of the SIFT vector around in order to create the complementary maxima (or minima) vocabulary. The process of inverting a set of cluster centroids is linear in the number of centroids (i.e. $O(K_2)$) and is thus obviously many times cheaper than performing a second clustering operation (irrespective of the algorithm used).

5. EXPERIMENTS AND DISCUSSION

The previous section motivates a number of theoretical arguments showing how the time taken for clustering can be reduced, but it is important to experimentally validate that when performing these optimisations the quality of the visual terms themselves is not diminished with respect to the task at hand. In particular, under the assumption that we want to use the visual terms in a large-scale retrieval scenario, we set out to investigate:

1. What is the retrieval performance of using either minimum features or maximum features alone?
2. What happens to the retrieval performance if we use a vocabulary learnt from minimum features to quantise maximum features?
3. What is the retrieval performance differential if we only learn a vocabulary on minimum features and then artificially invert it to create a maximum vocabulary?
4. What is the retrieval performance differential between learning separate vocabularies using minima and maxima and then combining them, versus just learning a single vocabulary across all the features?

5.1 Experimental setup

The experiments performed to investigate the vocabulary construction parameters take the form of a traditional image retrieval or object recognition experiment. The UKBench dataset[1] and evaluation protocol is used as the basis for the experiments presented here; the UKBench dataset consists of 10200 images of 2550 specific objects under varying orientation and illumination conditions. There are 4 images of each object in the dataset. The UKBench retrieval protocol is to take each image in turn as a query and calculate the four best matches (one, usually the first, of which should be the query image itself). A score is assigned based on how many of the top-four images are of the same object as the query. The score is averaged over all 10200 queries, and has a maximum value of 4.

Vocabulary Datasets.

In a retrieval system, the optimal way to create a visual term vocabulary would be to cluster all of the features of the images to be stored inside the system. However, in the real world, corpuses are often dynamic and using all the features is not possible. In the worst possible scenario, the dataset to be indexed may be unknown and thus the vocabulary may have to be learned from a completely different set of source images. In order to model the first of these two extremes, we have created vocabularies using all of the features we extract from the UKBench dataset. To model the second extreme, we use an entirely different non-overlapping dataset. Specifically, for the second extreme we learn our vocabularies from the MIRFLICKR-25000 dataset[2] [5]. The MIRFLICKR-25000 (referred to as *MIRFlickr* for the remainder of the paper) dataset consists of 25000 high-quality photos downloaded from Flickr that were rated to have a high *interestingness*[3]. Over both datasets we have created a large range of different sized vocabularies to assess the effect of vocabulary size on retrieval performance. The vocabularies were learned using all the features of the respective datasets (over 14 million for the UKBench and over 20 million from MIRFLICKR); see Table 1 for exact numbers of minima/maxima features.

Interest region detector.

Our experiments use a difference-of-Gaussian interest region detector to find interest regions which are then described with SIFT descriptors. Our detector/descriptor code has equivalent performance to the binary available from Lowe[4], but is implemented in Java and operates as a Map-Reduce program on a Hadoop cluster in order to efficiently batch process large image corpora [5]. In addition, our implementation has been modified to extract an extra variable for each interest region that describes whether the region was generated from local maxima or minima of the difference-of-Gaussian.

[1] http://www.vis.uky.edu/~stewe/ukbench/

[2] http://press.liacs.nl/mirflickr/

[3] see http://www.flickr.com/explore/interesting/

[4] http://www.cs.ubc.ca/~lowe/keypoints/

[5] More information will be made available from http://www.openimaj.org

k-means implementation.

For clustering we use an efficient multithreaded implementation of approximate k-means implemented in Java. Where possible, we keep the data, kd-trees and cluster centroids in memory, but if the data is too large it is read in batches from disk. Following the parameters specified in [13, 12], our k-means implementation is set to perform 30 iterations, and we use an ensemble of 8 randomised BBF kd-trees.

Retrieval testbed implementation.

We use a custom test-harness that is designed specifically for performing experiments with the UKBench dataset and protocol. In addition to generating UKBench scores, interpolated precision-recall curves are also generated. The test-harness is backed by a retrieval engine that indexes the quantised features using a highly-compressed inverted index and lexicon. During querying the lexicon is maintained in memory, but the inverted index is accessed directly from disk as required. Many different types of distance metric can be specified at run-time for the retrieval engine, but for the experiments presented here we restrict the distance metrics to the L1 distance and an IDF (inverse-document-frequency) weighted L1 distance (L1IDF).

Formally, defining q_i as the number of occurrences of the i-th term in the query and d_i as the number of occurrences of the i-th term of a database entry, then we define:

$$\hat{q}_i = q_i w_i$$
$$\hat{d}_i = d_i w_i$$

where w_i represents a weighting for the term. The L1 distance is defined as:

$$L_1(\hat{q}, \hat{d}) = ||\frac{\hat{q}}{||\hat{q}||} - \frac{\hat{d}}{||\hat{d}||}||$$

In the case of the unweighted L1 distance, w_i is set to 1. In the case of the IDF-weighted distance, w_i is defined as:

$$w_i = \ln \frac{N}{N_i}$$

where N is the total number of images in the corpus, and N_i is the number of images that contain visual term i.

5.2 Quantisation speedup

Before we discuss retrieval results, we first must check that the theoretical improvements in clustering time suggested in Section 4 actually hold. Figure 4 shows the actual time taken to cluster two sets of SIFT features from the UKBench dataset using AKM. The first set contains 15 million features, and the second set is half of that size. Each point on the graph was created by averaging the time taken to build the respective vocabulary over three runs. The error bars are +/- two standard deviations of the times from the individual runs. The times were calculated on a machine with 16 processor cores (and thus the clustering software used 16 threads).

The figure clearly shows that our theoretical assumptions are validated; in particular, the cost of performing two clusterings of the smaller dataset with half the vocabulary size takes less time than performing a single clustering of the large dataset with a full vocabulary size. If the vocabulary inversion technique can be used, then only a single clustering of the smaller dataset with half the vocabulary size need be performed, and the gains are even greater.

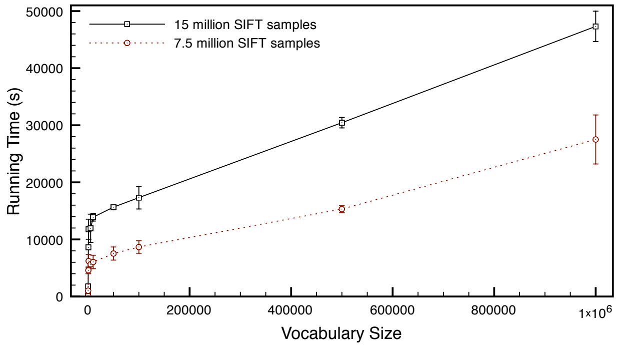

Figure 4: Actual time taken to create different sized vocabularies.

5.3 Retrieval results

5.3.1 *Baseline retrieval results*

Baseline UKBench retrieval results using singular vocabularies of a range of different sizes are shown in Figure 5. The vocabularies were trained on all the features from the respective datasets (over 14 million SIFT features for UKBench and over 20 million for MIRFlickr). The graph shows a number of interesting features which warrant discussion.

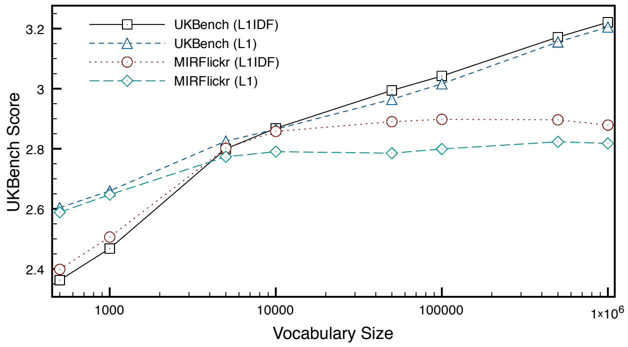

Figure 5: Baseline retrieval scores using all features without separation

Firstly, we will discuss the trends seen with vocabulary sizes in excess 5000-10000 terms. As vocabulary size increases above 10000 terms, the features learned from the UKBench dataset perform much better than those learned from MIRFlickr. This is not an unexpected result, as the UKBench images should be better modelled by a vocabulary learnt directly from their own feature space. The MIRFlickr vocabulary attains a peak retrieval performance score of about 2.9 with 100000 terms, after which the retrieval performance slowly begins to decay. Whilst not shown in this graph, in other experiments we have found that the best possible score with a vocabulary learned from the UKBench data gives a peak score of 3.29 with a vocabulary of 4 million terms. The peak in a graph of vocabulary size versus retrieval performance occurs because small vocabularies are over-generalised and lead to mismatches, whilst large vocabularies are over-specialised and lead to similar features being assigned to different terms. Above 10000 terms the IDF weighted distance measure consistently performs better than the unweighted variant, although the difference is much greater for the MIRFlickr vocabulary.

At vocabulary sizes of less than 5000 terms the graph tells a very different story. The unweighted L1 distance gives much larger scores than the weighted variant as vocabulary size decreases. Also, the scores attained with the two different vocabulary training sets are very similar. This suggests that the spaces occupied by SIFT features from both MIRFlickr and UKBench share a common set of core features that can be effectively extracted by k-means. In turn, this also suggests that a universal vocabulary that can be applied to any image set might well be possible for small vocabulary sizes at least.

In our experiments we are mostly interested in larger vocabularies which attain higher retrieval performance scores, so in the remainder of this paper we will only present scores calculated using the L1IDF distance measure.

5.3.2 *Separate minimum and maximum features*

Table 2 shows the UKBench scores for four different vocabulary sizes when using either the minima and maxima separately. In all cases, the scores for the minima are almost the same as for maxima. Using only the minimum (or maximum) features gives a lower score than the baseline (using all features) for the same vocabulary size (c.f. Figure 5). This indicates that the minimum and maximum features within an image are complementary to each other.

Table 2: Retrieval scores for separated minimum and maximum features

Voc.	UKBench Quantiser			MIRFlickr Quantiser		
Size	*Base*	Min	Max	*Base*	Min	Max
500	*2.36*	2.26	2.23	*2.40*	2.28	2.29
5000	*2.80*	2.64	2.64	*2.80*	2.64	2.62
50000	*2.99*	2.83	2.84	*2.89*	2.70	2.66
500000	*3.17*	3.06	3.07	*2.90*	2.65	2.65

5.3.3 *Maximum features quantised with a minimum vocabulary*

Figure 6 shows the effect on retrieval scores when maximum features are quantised using clusters learnt from the minimum features, compared to clusters learnt from maximum features. The figure shows clearly that quantising maximum features with a vocabulary learnt from minimum features leads to relatively poor performance. The obvious hypothesis for the drop in performance is that the feature space occupied by the minimum features is different to the space occupied by the maximum features; that is to say the SIFT space is bimodal as suggested in the previous section.

To get an idea of how different the two feature sets are, it is instructive to look at the statistics of the quantisation process, and see how many of the available visual terms from the minimum space are actually used when applied to the maximum space. Table 3 shows the actual number (and percentage) of terms used by maximum features when quantised using a minimum vocabulary. The similarity between the numbers from the UKBench and MIRFlickr datasets indicates that the SIFT feature spaces from these two different image collections share the same fundamental underlying morphology. The fact that relatively poor retrieval performance is attained at low vocabulary sizes, even though the vocabulary usage is high, is an indicator that the minimum and maximum feature spaces do not overlap (at least on the whole). This is further confirmed by the decreasing vocabulary usage as the vocabulary size increases.

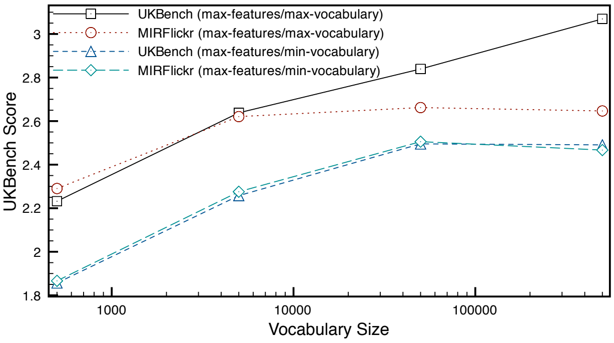

Figure 6: Retrieval performance of maximum features quantised with vocabularies learned from minimum features versus maximum features quantised using vocabularies learned from maxima features.

Table 3: Vocabulary usage when quantising maximum features with vocabularies learned from minimum features.

Voc. Size	UKBench Quantiser Vocabulary usage	MIRFlickr Quantiser Vocabulary usage
500	496 (99.2%)	495 (99.0%)
5000	4687 (93.7%)	4622 (92.4%)
50000	36859 (73.7%)	36695 (73.4%)
500000	199780 (40.0%)	206145 (41.2%)

5.3.4 *Maximum features quantised with an inverted minimum vocabulary*

Section 4.1 described how, by assuming the SIFT feature space is symmetric with respect to intensity inversion and thus minimum and maximum features, a vocabulary for maximum features could be created efficiently by automatically inverting a minimum vocabulary (or vice-versa). Figure 7 shows the retrieval performance of maximum features quantised using inverted minimum vocabularies, compared to maximum features quantised with vocabularies learnt from maximum features. With respect to the vocabularies learnt from the MIRFlickr dataset, the UKBench retrieval performance between the inverted-minimum vocabulary and maximum vocabulary is virtually indistinguishable. However, with the vocabularies trained using the UKBench data, there is a big retrieval performance differential as vocabulary sizes increase above 5000 terms. This differential indicates that the inverted-minimum vocabulary fails to capture all of the intricacies of the space actually spanned by the maximum features. Table 4 shows that the usage of the inverted minimum vocabularies is much more consistent with respect to vocabulary size than with the non-inverted vocabularies in Table 3.

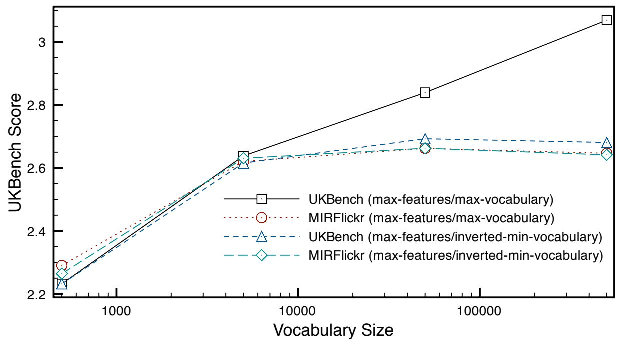

Figure 7: Retrieval performance of maximum features quantised with inverted vocabularies learned from minimum features versus maximum features quantised using vocabularies learned from maximum features.

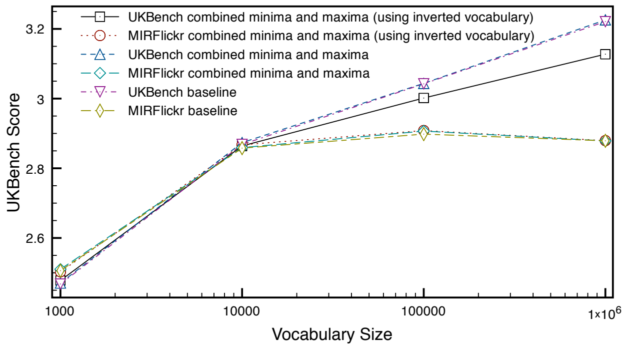

Figure 8: Retrieval performance against the baseline when minima and maxima are processed separately and then combined.

Overall, these results show that artificial vocabulary inversion is a plausible technique for efficiently creating a vocabulary if the vocabulary size is small, or the entire image corpus being indexed is not available at vocabulary creation time. Additionally, as shown in the following subsection, any performance differential between the inverted-minimum vocabulary and maximum vocabulary is partially made up when the minimum and maximum terms are combined.

5.3.5 Combined minima and maxima

The easiest way to combine pairs of visual term occurrence vectors from quantised minimum and maximum features is to concatenate the vectors together for each image. The overall vocabulary size is then equivalent to the sum of the minimum and maximum vocabulary sizes. Figure 8 shows the retrieval performance of terms from minimum and maximum vocabularies combined in this way. Maximum features created with both maximum vocabularies and inverted minima vocabularies are considered. For vocabularies learned from the MIRFlickr dataset, there is virtually no discernible difference in retrieval performance between the baseline (obtained by clustering both minimum and maximum features together) versus combinations of minimum and maximum terms. This applies to the case when the maximum features were quantised with vocabularies learned from maximum features (with half the number of total terms) and to the case where maximum features were quantised with inverted minimum vocabularies.

With respect to the vocabularies learnt from the UKBench data, again the results show that performing separate clusterings of minima and maxima with half of the total vocabulary size leads to equivalent retrieval performance to performing a single full-sized clustering of all the features. With the minimum terms combined with maximum terms quantised with inverted minimum vocabularies and a total vocabulary size of less than 10000 terms, the retrieval performance is equivalent to the baseline. Above 10000 terms, there is a drop in performance from the baseline, but the overall retrieval performance is still better than when using either set of terms by themselves.

Table 4: Vocabulary usage when quantising maximum features with inverted vocabularies learned from minimum features.

Voc. Size	UKBench Quantiser Vocabulary usage	MIRFlickr Quantiser Vocabulary usage
500	500 (100%)	500 (100%)
5000	5000 (100%)	5000 (100%)
50000	49985 (99.9%)	49988 (99.9%)
500000	464377 (92.9%)	485345 (97.1%)

6. CONCLUSIONS AND FUTURE WORK

This paper has shown how the intensity inversion characteristics of the SIFT descriptor and local interest region detectors can be exploited to decrease the time it takes to create vocabularies of visual terms. Through a large batch of experiments we have confirmed the theoretical ideas presented, and shown the effect on retrieval performance. In particular, we have shown clustering inverted and non-inverted (or minimum and maximum) features separately results in the same retrieval performance when compared to clustering all the features as a single set (with the same overall vocabulary size). Our experiments have also shown that minimum and maximum features are complementary to each other, and the best performance is achieved when they are used together. The experiments also show that the SIFT feature space is bimodal with respect to inverted and non-inverted SIFT features, and that the subspaces occupied by the non-inverted and inverted features are either non-overlapping, or minimally overlapping. Additionally, we have demonstrated that it is possible to artificially invert a vocabulary learned from minimum (or maximum) features to create a vocabulary for maximum (or minimum respectively) features without the cost of performing a clustering of the features. Whilst this work has concentrated on using k-means variants for vocabulary creation, other techniques such as Locality Sensitive Hashing would also benefit from the approach proposed here.

The retrieval experiments presented in the paper have confirmed the expected result that when a large visual vocabulary is learned over the set of images being retrieved, higher performance is achieved than if the vocabulary is learnt from a different dataset. However, with smaller vocabularies (of around 10000 terms or less), the variation between training sets for the vocabulary creation is essentially negligible. This suggests that is is possible to create a *universal* vocabulary for small vocabulary sizes at least.

This work has generated many ideas for future work. Firstly, we have made use of the inversion properties of the SIFT feature to make clustering more efficient, however, the produced visual terms are not themselves invariant to inversion. Inversion invariance of SIFT features has been achieved before by transforming the SIFT vectors [7]. However, using the ideas in this paper, inversion invariant visual terms could be constructed by creating a vocabulary from minimum features and using this vocabulary to quantise both minimum features and maximum features, but transforming the maximum features to the minimum space beforehand. Inversion-invariant visual terms could have potential use in a number of specific retrieval scenarios.

Secondly, there are a number of other ways in which the SIFT feature space can be considered to be bimodal and possibly symmetric; for example through mirroring of the image. All of the ideas in this work could probably be applied to mirrored features. The only problem with this is that information on whether or not a feature is mirrored is unlikely to be available directly from the interest region detector. However, it should be possible to derive a test at the image pixel level that determines the modality of the region.

Finally, our experiments have indicated that it appears to be possible to create a universal vocabulary of visual terms when the number of terms is relatively small. It would be pertinent to investigate whether a larger universal vocabulary is possible with a sufficiently large and varied training set. The techniques presented in this paper for improving the computational performance will certainly be helpful in investigating this.

7. ACKNOWLEDGMENTS

This work was supported by the European Union under the Seventh Framework project LivingKnowledge (Contract No. 231126), and the LiveMemories project, graciously funded by the Autonomous Province of Trento (Italy).

References

[1] J. D. Bustard and M. Nixon. Robust 2D Ear Registration and Recognition Based on SIFT Point Matching. In *BTAS 2008*, September 2008.

[2] J. Hare and P. Lewis. Automatically annotating the mir flickr dataset: Experimental protocols, openly available data and semantic spaces. In *MIR '10: Proceedings of the international conference on Multimedia information retrieval*, pages 547–556. ACM, March 2010.

[3] J. S. Hare and P. H. Lewis. On image retrieval using salient regions with vector-spaces and latent semantics. In W. K. Leow, M. S. Lew, T.-S. Chua, W.-Y. Ma, L. Chaisorn, and E. M. Bakker, editors, *CIVR*, volume 3568 of *LNCS*, pages 540–549, Singapore, 2005. Springer. ISBN 3-540-27858-3.

[4] J. S. Hare and P. H. Lewis. Content-based image retrieval using a mobile device as a novel interface. In R. W. Lienhart, N. Babaguchi, and E. Y. Chang, editors, *Proceedings of Storage and Retrieval Methods and Applications for Multimedia 2005*, pages 64–75, San Jose, California, USA, January 2005. SPIE.

[5] M. J. Huiskes and M. S. Lew. The MIR Flickr Retrieval Evaluation. In *MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2008. ACM.

[6] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, January 2004.

[7] R. Ma, J. Chen, and Z. Su. MI-SIFT: mirror and inversion invariant generalization for SIFT descriptor. In *CIVR*, pages 228–235, 2010.

[8] J. B. Macqueen. Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[9] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In P. L. Rosin and A. D. Marshall, editors, *BMVC*. British Machine Vision Association, 2002. ISBN 1-901725-19-7.

[10] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. In *CVPR*, volume 2, pages 257–263, June 2003.

[11] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, pages 2161–2168, 2006.

[12] J. Philbin. *Scalable Object Retrieval in Very Large Image Collections*. PhD thesis, University of Oxford, 2010.

[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.

[14] J. Sivic and A. Zisserman. Video google: A text retrieval approach to object matching in videos. In *ICCV*, pages 1470–1477, October 2003.

[15] J. Sivic, B. Russell, A. Efros, A. Zisserman, and W. Freeman. Discovering objects and their location in images. In *ICCV*, volume 1, pages 370 – 377 Vol. 1, 2005. doi: 10.1109/ICCV.2005.77.

[16] W. Zhou, Y. Lu, H. Li, Y. Song, and Q. Tian. Spatial coding for large scale partial-duplicate web image search. In A. D. Bimbo, S.-F. Chang, and A. W. M. Smeulders, editors, *ACM Multimedia*, pages 511–520. ACM, 2010. ISBN 978-1-60558-933-6.