How Many Bits Do I Need For Matching Local Binary Descriptors?

Conference Paper · May 2016

DOI: 10.1109/ICRA.2016.7487369

2 authors:

Pablo Fernández Alcantarilla
iRobot Corporation
40 PUBLICATIONS **353** CITATIONS
SEE PROFILE

Bjorn Stenger
University of Cambridge
66 PUBLICATIONS **2,063** CITATIONS
SEE PROFILE

How Many Bits Do I Need For Matching Local Binary Descriptors?

Pablo F. Alcantarilla Björn Stenger

***Abstract*— In this paper we provide novel insights about the performance and design of popular pairwise tests-based local binary descriptors with the aim of answering the question: *How many bits are needed for matching local binary descriptors?* We use the interpretation of binary descriptors as a Locality Sensitive Hashing (LSH) scheme for approximating Kendall's tau rank distance between image patches. Based on this understanding we compare local binary descriptors in terms of the number of bits that are required to achieve a certain performance in feature-based matching problems. Furthermore, we introduce a calibration method to automatically determine a suitable number of bits required in an image matching scenario. We provide a performance analysis in image matching and structure from motion benchmarks, showing calibration results in visual odometry and object recognition problems. Our results show that excellent performance can be achieved using a small fraction of the total number of bits from the whole descriptor, speeding-up matching and reducing storage requirements.**

I. Introduction

Local image features are widely used in applications such as object recognition, image registration, and Structure from Motion (SfM). The principle is to detect repeatable image keypoints, compute distinctive local image descriptors, and match these between different images. Traditional vector-based descriptors in floating-point representation such as SIFT [1], SURF [2] and KAZE [3] achieve good performance in most applications. However, the price to pay is the amount of memory and time required for storing and matching these descriptors.

Recently there has been significant interest in local binary descriptors [4], [5], [6], [7], [8]. With the proliferation of camera-enabled mobile devices with limited computational resources, new binary descriptors have appeared that aim to reduce computational complexity while retaining the performance of vector-based descriptors. Binary descriptors are compared using the Hamming distance, which is particularly efficient on dedicated bit-counting hardware. In addition, the storage requirements are significantly lower compared to vector-based descriptors, enabling large-scale applications [9], [10]. Furthermore, binary descriptors are widely used in robotics in applications such as visual Simultaneous Localization and Mapping (SLAM) [11] and place recognition for detecting loop closure [12], [13].

The most extensive recent evaluation of binary descriptors by Heinly *et al*. [14] evaluates descriptors on an extended version of the Oxford benchmark dataset [15]. Recently, a similar evaluation focusing on mobile visual search applications has been carried out in [16]. These evaluations of binary descriptors evaluate the descriptors in a similar way to vector-based descriptors. In [17] it was shown that the BRIEF descriptor [5] and similar binary descriptors based on pairwise intensity comparisons can be interpreted as a Locality Sensitive Hashing (LSH) approximation for Kendall's tau distance [18]. Kendall's tau distance is a metric on ranked lists, which counts the number of pairwise disagreements between the two lists. It is commonly used in document information retrieval, while here we apply it to measure distances between image patches. We show that the Hamming distance between binary descriptors is an unbiased estimate of Kendall's tau metric, we derive expressions for its expectation and variance, and compare binary descriptors in terms of the number of bits required to achieve certain performance targets.

In most matching applications, the full length of the binary descriptor is used by default. We show that this is not necessary and that excellent performance can be achieved by using a small number of bits, implying significant benefits in terms of memory requirements and matching speed. Using this evaluation, researchers can select the number of bits for a desired level of performance in a particular application. In particular, we consider experiments on benchmark datasets for image matching and SfM. We introduce a novel calibration method to automatically determine the number of bits required in an image matching scenario, showing experimental results in Visual Odometry (VO) and object recognition tasks.

Our analysis includes new up-to-date binary descriptors, in particular FREAK [4], BinBoost [19], LDB [8] and M-LDB [20]. The discussion focuses on local binary descriptors that are computed directly from pairwise tests. Other descriptors that require supervised learning such as [21], [22] are outside the scope of this paper.

II. Local Binary Descriptors

Local binary descriptors are built from a set of pairwise intensity comparisons near the point of interest. Each bit in the descriptor is the result of exactly one comparison. Most binary descriptors differ in the spatial sampling pattern and the descriptor length. The sampling pattern can be fixed or adapted to obtain descriptors invariant to scale and rotation. The set of pairwise comparisons is not limited to intensity values; gradients and other image cues can also be used to increase the discriminative power of the descriptors [8], [20].

Pablo F. Alcantarilla is with iRobot Corporation, 10 Greycoat Place, Victoria, London, UK. palcantarilla@irobot.com. Björn Stenger is with Rakuten, Inc. bjorn@cantab.net. This work was done when both authors were at Toshiba Research Europe, Cambridge, UK.

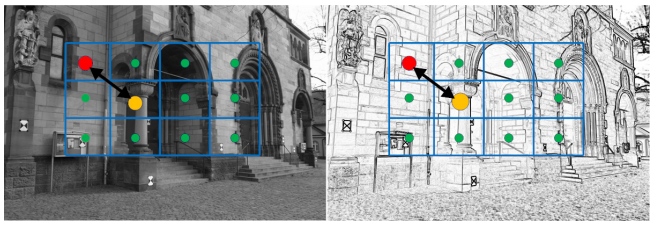


Fig. 1: ***Binary descriptor computation.*** *Given an image patch* $\mathbf{p}$ *and a sampling pattern, one bit in the descriptor results from the comparison of image intensities (left) or other image cues such as gradients (right). The sampling pattern can be either fixed or adapted to the dominant orientation and scale.*

Considering a smoothed vectorized image patch $\mathbf{p}$, a binary test $\varphi(\cdot)$ is defined as

$$\varphi\left(\mathbf{p};\mathbf{x},\mathbf{y}\right) := \begin{cases} 1\,, & \text{if } f(\mathbf{x}) < f(\mathbf{y})\,,\ \mathbf{x} \neq \mathbf{y} \\ 0\,, & \text{otherwise} \end{cases}, \quad (1)$$

where $f(\mathbf{x})$ is a function that returns an image feature response for a certain pixel $\mathbf{x} = (x, y)$ in the image patch $\mathbf{p}$. This function $f(\mathbf{x})$ can simply be the smoothed image intensity I at one pixel location, as proposed in [4], [5], [6], [7]. Additionally, the function $f(\mathbf{x})$ can also be the concatenation of different binary comparisons such as averaged image intensities $\hat{I}$ or image gradients L_x, L_y on a particular grid cell $\mathbf{c_i}$ in the image patch $\mathbf{p}$, as proposed recently in [20], [8]. In this case each binary test returns a single bit per pairwise comparison between similar channels

$$f(\mathbf{x}) := \left\{\hat{I}(\mathbf{c_i}), L_x(\mathbf{c_i}), L_y(\mathbf{c_i})\right\}. \quad (2)$$

Finally, the resulting descriptor $d_n(p)$ is computed as a vector of n binary tests

$$d_n(\mathbf{p}) := \sum_{1\leq i\leq n} 2^{i-1}\varphi\left(\mathbf{p};\mathbf{x}_i,\mathbf{y}_i\right), \quad (3)$$

where the number of binary tests n is typically determined empirically. Fig. 1 shows one example of a binary test in an image patch, in which intensity and gradient comparisons are considered.

A. Review of Local Binary Descriptors

This section briefly reviews recent binary descriptors, which are evaluated in this paper. The BRIEF descriptor [5] uses a rectangular sampling pattern from which the sampling points are selected randomly from an isotropic Gaussian distribution centered at the feature location. However, BRIEF is not invariant to rotation or scale changes.

The BRISK descriptor [6] uses a symmetric sampling pattern, where sample points are located in concentric circles around the feature location. In the ORB [7] and FREAK [4] descriptors the set of pairwise comparisons is learned from training data. In contrast to BRIEF, these descriptors are invariant to rotation and scale changes.

The LDB descriptor [8] increases the discriminative power by including multiple image cues. It computes a binary string based on binary tests of intensity and gradient differences on grid cells within an image patch. Grids at different spatial resolutions are used to increase the discriminative power. The pairwise comparisons are selected either randomly or by entropy-based bit selection. The main drawback of LDB is that it is neither rotation nor scale invariant.

The M-LDB descriptor [20] is based on LDB, adding rotation and scale invariance. While in the LDB descriptor the mean intensities and derivatives are computed with integral images, in M-LDB the intensities and derivatives are obtained by sampling in the image scale space. Image derivatives are computed with Scharr filters, which approximate rotation invariance significantly better than other filters [23]. In summary, descriptors either use random bit selection (BRIEF, LDB, M-LDB), learn the pairwise comparisons (ORB, FREAK, LDB) or use a fixed pattern (BRISK). Interestingly, BRIEF and LDB [5], [8] reported superior matching results using a random bit selection scheme when applied to matching in the Oxford benchmark dataset of images with no geometric transforms.

III. Interpretation of Binary Descriptors as LSH for Kendall's tau Rank Distance

The goal of binary descriptors extracted from image patches is to obtain distinctive binary codes that can be used efficiently in image matching applications. Similar image patches should have similar binary codes with a small Hamming distance; dissimilar image patches should lead to binary codes with larger Hamming distances. This goal is the same as in binary LSH schemes [24], [25] that approximate distance computation by first applying hash functions to larger data vectors and then computing the Hamming distance between the resulting binary vectors. The goal in LSH is to map similar objects to similar hash codes with high probability. In contrast, local binary descriptors directly build short binary descriptors by comparing the intensities of pairs of points without ever creating a long descriptor.

A. Kendall's tau Rank Distance

Kendall's tau rank distance is a metric that counts the number of pairwise disagreements between two ranking lists [18]. This metric is widely used in information retrieval, where one is often faced with the problem of computing the similarity or correlation between two ranked lists of elements [26], [27]. Given two input data vectors $\mathbf{p}_1$ and $\mathbf{p}_2$ of dimension d, the normalized Kendall's tau distance between these two data vectors is defined as

$$d_\tau(\mathbf{p}_1,\mathbf{p}_2) = \frac{1}{\binom{d}{2}}\sum_{i<j} k\left(\left(\mathbf{x}_i,\mathbf{y}_j\right)_{\mathbf{p}_1},\left(\mathbf{x}_i,\mathbf{y}_j\right)_{\mathbf{p}_2}\right). \quad (4)$$

The function $k\left(\left(\mathbf{x}_i,\mathbf{y}_j\right)_{\mathbf{p}_1},\left(\mathbf{x}_i,\mathbf{y}_j\right)_{\mathbf{p}_2}\right)$ is a symmetric kernel function that returns 1 if the rankings are in disagreement, *i.e.* $\pi(\mathbf{x}_i)_{\mathbf{p}_1} > \pi(\mathbf{y}_j)_{\mathbf{p}_1}$ and $\pi(\mathbf{x}_i)_{\mathbf{p}_2} < \pi(\mathbf{y}_j)_{\mathbf{p}_2}$ or the opposite, and 0 if the rankings are in agreement. The function $\pi(\mathbf{x}_i)_{\mathbf{p}}$ represents the ranking of the element $\mathbf{x}_i$ in a particular image patch $\mathbf{p}$.

B. LSH for Kendall's tau Rank Distance

Let us consider a family of hash functions $\mathcal{H}$ operating on a collection of image patches $\mathbf{p}$ of dimension d and sampling pattern of $\binom{d}{2}$ pairs. Each hash function $h(\mathbf{p}) \in \mathcal{H}$ selects one independent and random pairwise comparison (i, j) from the sampling pattern, returning 1 if $\pi(\mathbf{x}_i) > \pi(\mathbf{y}_j)$ and 0 otherwise. For a distance threshold r and an approximation factor c, the family of hash functions $\mathcal{H}$ is called (r, cr, P_1, P_2)-sensitive if for any two image patches $\mathbf{p}_1$ and $\mathbf{p}_2$:

- if $\mathbf{p}_1 \in B(\mathbf{p}_2, r)$, then $Pr_{\mathcal{H}}\left[h(\mathbf{p}_1) = h(\mathbf{p}_2)\right] \geq P_1$
- if $\mathbf{p}_1 \notin B(\mathbf{p}_2, cr)$, then $Pr_{\mathcal{H}}\left[h(\mathbf{p}_1) = h(\mathbf{p}_2)\right] \leq P_2$,

where $B(\mathbf{p}_2, r)$ represents a ball with distance r centered at $\mathbf{p}_2$. In order to show that the family of hash functions $\mathcal{H}$ is LSH for the Kendall's tau rank distance, we require that $P_1 > P_2$ [28]. Observe that the probability $Pr_{\mathcal{H}}\left[h(\mathbf{p}_1) = h(\mathbf{p}_2)\right]$ is equal to the fraction of coordinates in which the rankings of $\mathbf{p}_1$ and $\mathbf{p}_2$ agree. Therefore, $P_1 = 1 - r/d_p$, while $P_2 = 1 - cr/d_p$, where $d_p = \binom{d}{2}$. As long as the approximation factor c is greater than 1, we have that $P_1 > P_2$ and the property of LSH holds for $\mathcal{H}$ over the Kendall's tau metric.

C. Relation between Binary Descriptors and Kendall's tau Rank Distance

Let us consider the binary descriptors from Section II applied to two vectorized image patches $\mathbf{p}_1$ and $\mathbf{p}_2$ of dimension d and a sampling pattern consisting of all possible pairwise comparisons $\binom{d}{2}$. The normalized Hamming distance between the two binary descriptors is

$$d_H(\mathbf{p}_1, \mathbf{p}_2) = \frac{1}{\binom{d}{2}} \sum_{i<j} \mathbb{1}\left(\varphi\left(\mathbf{p}_1; \mathbf{x}_i, \mathbf{y}_j\right) \neq \varphi\left(\mathbf{p}_2; \mathbf{x}_i, \mathbf{y}_j\right)\right). \tag{5}$$

where the function $\mathbb{1}(\cdot)$ is the indicator function that returns 1 if the output of the binary tests $\varphi\left(\mathbf{p}_1; \mathbf{x}_i, \mathbf{y}_j\right)$ and $\varphi\left(\mathbf{p}_2; \mathbf{x}_i, \mathbf{y}_j\right)$ for a particular location (i, j) is different.

Clearly, the Hamming distance between two binary descriptors is related to Kendall's tau distance in Eq. 4, where the binary test φ corresponds to the kernel function. A binary descriptor such as BRIEF and those based on intensity pairwise comparisons are effectively an LSH scheme approximating Kendall's tau distance between image patches. Binary descriptors that use multiple cues in their pairwise comparisons, such as LDB and M-LDB, can be interpreted as sums of Kendall's tau rank distance for each individual cue.

1) Expectation and Variance of Kendall's tau Approximation: The normalized Kendall's tau rank distance is a measure of the probability of disagreement between two ranked lists. Since the family of hash functions $h \in \mathcal{H}$ is an LSH scheme on Kendall's tau distance, we can compute the probability of disagreement between two ranking lists as [24]:

$$Pr\left[h(\mathbf{p}_1) \neq h(\mathbf{p}_2)\right] = d_\tau(\mathbf{p}_1, \mathbf{p}_2). \tag{6}$$

The Hamming distance between two image patches $\mathbf{p}_1$ and $\mathbf{p}_2$ under $\mathcal{H}$ follows a binomial distribution with parameters $\mathcal{B}(K, d_\tau(\mathbf{p}_1, \mathbf{p}_2))$, the expectation of the Hamming distance between two binary hash codes is

$$\mathbb{E}[d_H\left(h(\mathbf{p}_1), h(\mathbf{p}_2)\right)] = K d_\tau(\mathbf{p}_1, \mathbf{p}_2). \tag{7}$$

That is, the expectation of the Hamming distance between two binary hash codes of two image patches $\mathbf{p}_1$ and $\mathbf{p}_2$ is an unbiased estimate of Kendall's tau distance between them up to a constant scale factor K. Then the variance of the normalized Hamming distance can be shown to satisfy

$$\mathrm{Var}\left[\frac{1}{K} d_H\left(h(\mathbf{p}_1), h(\mathbf{p}_2)\right)\right] = \frac{d_\tau(\mathbf{p}_1, \mathbf{p}_2)}{K}\left(1 - d_\tau(\mathbf{p}_1, \mathbf{p}_2)\right). \tag{8}$$

The expression on the right shows that we require a certain number of K bits to approximate Kendall's tau distance with a small variance. The number of bits that are required to approximate Kendall's tau distance with a certain accuracy will be different for each local binary descriptor and each matching scenario.

IV. How to Calibrate your Binary Descriptor

In this section we introduce a calibration method to automatically find a suitable number of bits in a particular image matching scenario.

The *root mean squared relative error* (RMSRE) measures the relative error between the *target* and the *estimate*. In our case, the target is Kendall's tau similarity between two image patches (d_{τ_i}) and the estimate is the approximation performed by the Hamming distance of two binary descriptors ($\tilde{d}_{\tau_i}$). According to Eq. 7, we just need to divide the Hamming distance between two binary descriptors by K in order to obtain an estimate of Kendall's tau similarity. For a set of M samples, the RMSRE is defined as

$$\mathrm{RMSRE} = \left(\frac{1}{M}\sum_{i=1}^{M} ||d_{\tau_i} - \tilde{d}_{\tau_i}||^2 / ||d_{\tau_i}||^2\right)^{\frac{1}{2}}. \tag{9}$$

Each binary descriptor may exhibit different RMSRE for different patches and image transformations. However, regardless of the error in the approximation of Kendall's tau similarity, when the variance of the normalized Hamming distance (see Eq.8) is very small, the ranking of the descriptors does not change significantly. In order to have a common measure to compare and evaluate different binary descriptors, we introduce an error measure called *Binary Descriptor Approximation Error* (BDAE) that combines the RMSRE in the approximation of Kendall's tau similarity and the expectation of the variance of the normalized Hamming distance:

$$\mathrm{BDAE(K)} \;=\; 100 \cdot \mathrm{RMSRE} \cdot \mathbb{E}\left[\mathrm{Var}\left[\frac{d_H}{K}\right]\right]. \tag{10}$$

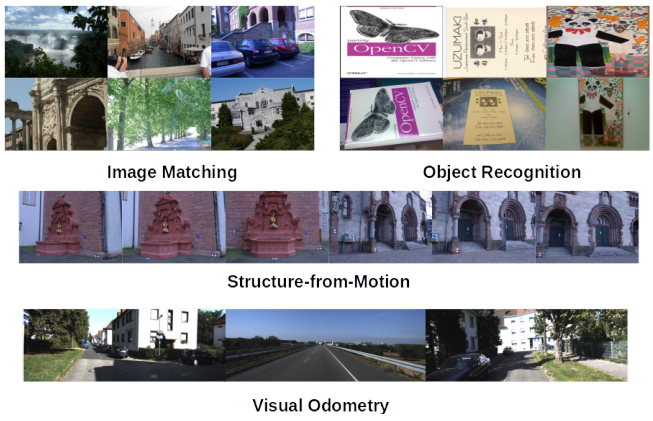


Fig. 2: ***Datasets used in our experiments.*** *Sample images from the datasets used in our experiments for image matching, SfM, VO and object recognition problems.*

V. Experimental Results

We use the interpretation of binary descriptors as LSH to approximate Kendall's tau distance between image patches to carry out an analysis in terms of the number of bits that are required to achieve a certain performance in standard descriptor-based matching problems in Section V-A. We then show results of our calibration method to automatically determine a suitable number of bits required in VO and object recognition problems in Section V-B.1 and V-B.2 respectively. Fig. 2 depicts some sample images from the datasets used in our experiments.

A. Effect of the Number of Bits in Local Binary Descriptors

In our analysis we test the performance of BRIEF, ORB, BRISK, FREAK, LDB and M-LDB in image matching experiments on a subset of the image matching benchmark proposed in [14] and SfM [29]. In addition, we compare the performance of binary descriptors to two standard vector-based descriptors (SURF, SIFT) and to BinBoost [19], which is a binary descriptor that requires supervised learning.

We analyze the performance of different local binary descriptors considering a common feature detector. In particular, we detect keypoints by finding local extrema of the multi-scale determinant of Hessian operator in a Gaussian scale space. As mentioned in [14], combining detectors and descriptors that are scale-invariant is not trivial, since each detector and descriptor uses its own definition for the scale of a feature. In our evaluation, we adapt the scale of each detected keypoint to an appropriate value as defined in the original implementation of each descriptor. Regarding orientation estimation, we use the orientation estimate provided by each descriptor.

Given the detected keypoints in a reference and a query image, we compute descriptors and obtain a set of *putative matches* using the *Nearest Neighbor Distance Ratio* (NNDR) strategy. This test compares the ratio distances between the two best matches for a given keypoint and accepts the match if the distance ratio is below 0.8. This is the same matching strategy used in [14].

For all binary descriptors we start from an initial descriptor length and increase the length of the descriptor in logarithmic steps to the maximum descriptor size as defined in their original implementations. For choosing the number of bits in the descriptor, we perform LSH and randomly choose K bits from the descriptor each time. We carry out 10 experiments per random bit selection and average the results. We also show averaged BDAE results, considering 10 experiments per random bit selection in the LSH approximation performed by the binary descriptors.

Our evaluation uses the OpenCV v3.0 implementations of SIFT, SURF, BRIEF, ORB, BRISK and FREAK. For LDB, M-LDB and BinBoost we use the original implementations provided by the authors.

1) ***Image Matching****:* Our image matching evaluation includes a subset of the sequences from the extended Oxford benchmark presented in [14]. The dataset includes several image sets with different geometric and photometric transformations such as image blur, lighting, viewpoint, scale changes, zoom, rotation, and JPEG compression. In addition, the ground truth homographies between reference and query images are also available. We also use the *Iguazu* dataset [3] for the image matching evaluation in the presence of Gaussian noise.

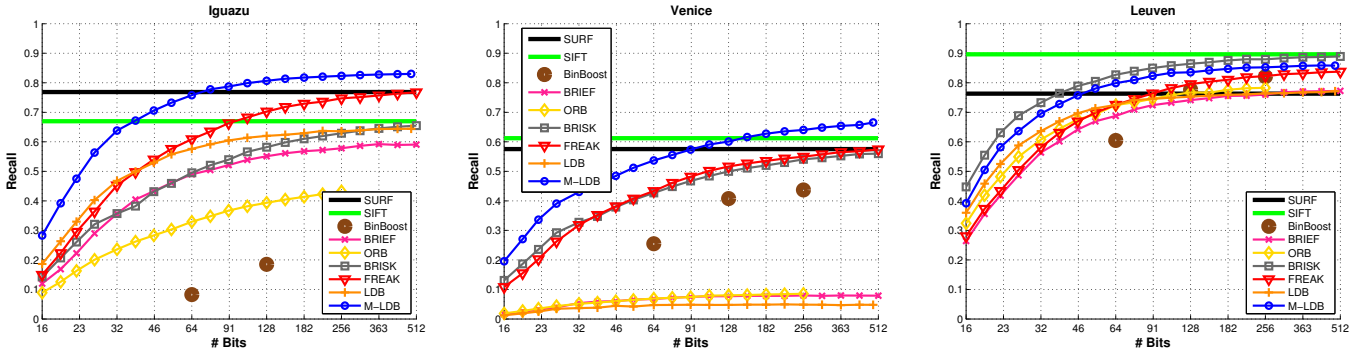
We evaluate local binary descriptors performance by computing *recall* versus the number of bits in the descriptor. Recall is computed as:

$$recall = \frac{\#correct\ matches}{\#correspondences}. \tag{11}$$

As defined in [15], recall measures how many of the possible correct matches were actually found. The number of correspondences is the number of matches that should have been identified given the keypoint locations in both images. For computing the number of correct matches we geometrically verify the correspondences using the ground truth information. The detected keypoints from the reference image are projected into the query image. The error in relative point location for two corresponding regions has to be less than 2.5 pixels as also used in [14].

We test the performance of the binary descriptors in their *upright* form, i.e. without rotation invariance. By avoiding the computation of the descriptor with respect to a dominant orientation, we can obtain increased discriminative power in those sequences where rotation invariance is not required. BRIEF and LDB are not rotation invariant, and therefore are always computed in their upright form. We use the upright version of SURF, SIFT, BinBoost, ORB, BRISK, FREAK and M-LDB in the *Iguazu*, *Venice*, *Leuven*, *Trees* and *UBC* datasets. We compute average results for all images in each sequence.

Fig. 3 shows the *recall* versus *number of bits* graphs for the image matching evaluation. We also show image matching statistics for each sequence such as: maximum recall, the number of bits K^* for which $\mathrm{BDAE} = 0.01$, and the number of bits needed to match the performance of SURF and SIFT.



Iguazu	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.59	0.57 / 223	--	--
ORB	0.43	0.42 / 192	--	--
BRISK	0.65	0.65 / 427	--	--
FREAK	**0.76**	0.75 / 264	**512**	**91**
LDB	0.64	0.63 / 190	--	--
M-LDB	**0.83**	0.82 / 197	**64**	**38**

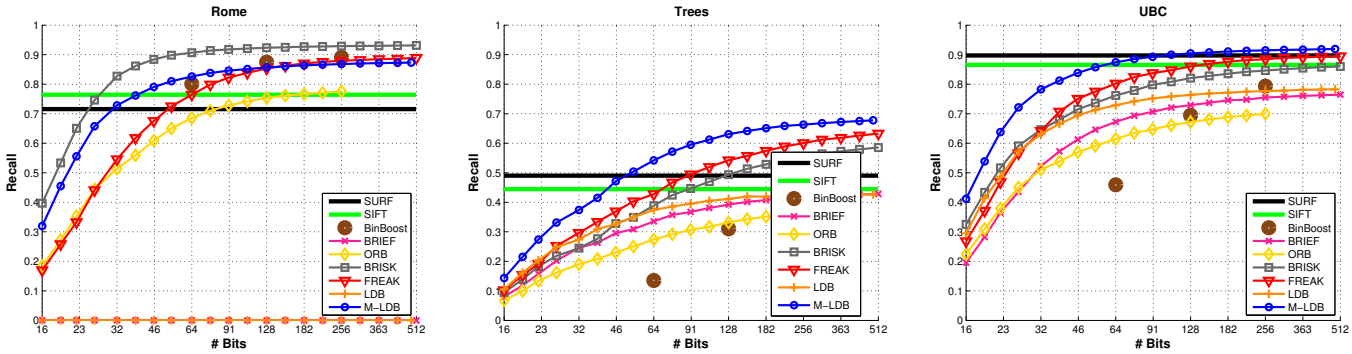
(a)

Venice	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.08	0.08 / 207	--	--
ORB	0.08	0.08 / 185	--	--
BRISK	0.56	0.55 / 329	512	--
FREAK	**0.57**	0.55 / 258	**512**	--
LDB	0.08	0.08 / 312	--	--
M-LDB	**0.66**	0.63 / 201	**91**	**152**

(b)

Leuven	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.77	0.74 / 223	215	--
ORB	0.78	0.77 / 216	128	--
BRISK	**0.89**	0.87 / 365	**38**	--
FREAK	0.84	0.82 / 274	90	--
LDB	0.77	0.75 / 198	215	--
M-LDB	**0.86**	0.84 / 197	**46**	--

(c)



Rome	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.00	0.00 / 197	--	--
ORB	0.77	0.77 / 238	76	181
BRISK	**0.93**	0.93 / 454	**26**	**26**
FREAK	**0.89**	0.88 / 354	**53**	**64**
LDB	0.00	0.00 / 171	--	--
M-LDB	0.87	0.86 / 209	32	38

(d)

Trees	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.43	0.41 / 223	--	--
ORB	0.36	0.35 / 192	--	--
BRISK	0.58	0.57 / 427	128	91
FREAK	**0.63**	0.61 / 264	**91**	**76**
LDB	0.43	0.41 / 223	--	--
M-LDB	**0.68**	0.67 / 197	**53**	**36**

(e)

UBC	Max. RC	RC / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	0.77	0.75 / 240	--	--
ORB	0.71	0.69 / 208	--	--
BRISK	0.86	0.85 / 383	--	512
FREAK	**0.89**	0.88 / 296	**362**	**128**
LDB	0.78	0.77 / 221	--	--
M-LDB	**0.92**	0.91 / 226	**107**	**64**

(f)

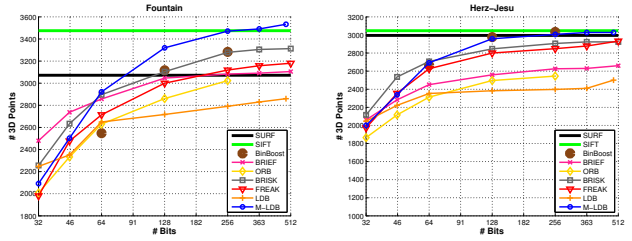
Fig. 3: ***Image matching evaluation****. Recall (RC) versus number of bits graphs and statistics for the image matching evaluation. (a) Iguazu (Gaussian noise), (b) Venice (scale), (c) Leuven (lighting), (d) Rome rotation), (e) Trees (blur), (f) UBC (JPEG compression) sequences. Best two global image matching statistics are in bold.*

2) ***Structure from Motion****:* We evaluate binary descriptors in a practical SfM application, using the *Fountain* and *Herz-Jesu* datasets from [30]. The *Fountain* dataset contains 6 images, while the *Herz-Jesu* one contains 7 images. We use the SfM pipeline from Bundler [29] to produce 3D reconstructions and evaluate binary descriptors with respect to the final number of reconstructed 3D points.

The number of reconstructed 3D points is an indicator of how good the descriptors are: A higher number of 3D points indicates that more descriptor matches have been geometrically validated in the SfM pipeline, and therefore descriptors are more discriminative. Fig. 4 depicts the *number of reconstructed 3D points* versus the *number of bits* and associated SfM statistics.

3) ***Discussion****:* As can be observed in most of the experiments, for each binary descriptor there is a certain number of bits for which recall, precision, or number of reconstructed 3D points saturate. The explanation of this behavior is that for that particular number of bits, the error in the variance and the error in the LSH approximation performed by the binary descriptors is small compared to the to Kendall's tau similarity variability. At this point we obtain little additional improvement in the descriptor ranking by adding more bits to the descriptor.

In the image matching experiments the number of bits for which recall saturates is normally below 100. In general, M-



Fountain	Max. 3D	# 3D / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	3105	3080 / 246	205	--
ORB	3023	2998 / 237	--	--
BRISK	**3313**	3286 / 286	**112**	--
FREAK	3180	3154 / 352	362	--
LDB	2860	2837 / 391	--	--
M-LDB	**3533**	3504 / 402	**128**	**362**

(a)

Herz-Jesu	Max. 3D	# 3D / K @BDAE = ε	# Bits SURF	# Bits SIFT
BRIEF	2660	2638 / 405	--	--
ORB	2545	2524 / 202	--	--
BRISK	2923	2899 / 242	--	--
FREAK	**2932**	2908 / 446	--	--
LDB	2500	2480 / 458	--	--
M-LDB	**3028**	3008 / 248	**128**	--

(b)

*Fig. 4: **SfM evaluation**. Number of reconstructed 3D points (3D) versus number of bits graphs and statistics for the SfM evaluation. (a) Fountain, (b) Herz-Jesu. Best two global SfM statistics are in bold.*

LDB obtains higher recall than the other binary descriptors for all numbers of bits. One reason for this behavior is that M-LDB also uses binary tests considering gradient information, and therefore the bits in the descriptor are less correlated than those descriptors that use intensity only. After M-LDB, FREAK is a close second, followed by BRISK and ORB. For those datasets where there is no change in rotation and scale, BRIEF and LDB exhibit very good performance, with LDB performance being slightly higher.

In the SfM experiments, vector-based descriptors obtain slightly better results than binary descriptors, followed closely by M-LDB, BRISK and FREAK. In the *Fountain* dataset, M-LDB outperforms SIFT for bit sizes larger than 362 and almost matches SIFT performance in the *Herz-Jesu* dataset with the full descriptor length. The number of reconstructed 3D points tends to saturate for bit sizes in the 64–100 range for all binary descriptors. In datasets with perspective transforms (e.g. changes in viewpoint), vector-based descriptors tend to perform better than binary ones, since there is a higher variability between corresponding image patches than in non-geometric transforms. This is a similar conclusion as found in [14] where SIFT showed the best performance considering perspective transforms. BinBoost exhibits good performance in the SfM experiments, since the descriptor was trained using image patches from SfM datasets [31]. However, the descriptor does not generalize well to other image transformations such as noise or blur.

In general, M-LDB obtains better performance than its binary competitors in the three experiments. Note that in some of the experiments, especially image matching, the number of bits needed to match vector-based descriptors is quite small for some descriptors such as M-LDB, BRISK or FREAK. Furthermore, as shown in our experiments the BDAE is a good indicator of the performance of the descriptors. When BDAE is small (*e.g.* 0.01) the error in the approximation and the variance is small and therefore descriptors ranking do not change significantly by adding more bits. This behavior seems to be consistent for all the analyzed descriptors. This can have important consequences for large-scale applications and camera-enabled mobile devices since computing and matching vector-based descriptors is a time-consuming operation compared to binary ones and storage requirements are more demanding. In addition, the computation of binary descriptors can be speeded up since only few bits are necessary in most scenarios, and therefore there is no need to compute the whole descriptor.

B. Calibration Experiments

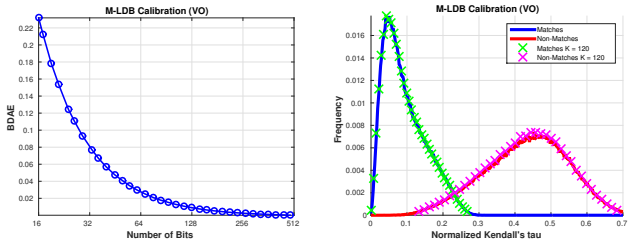
*1) **Stereo Visual Odometry**:* We test our calibration procedure considering the M-LDB descriptor in a feature-based stereo VO scenario using the KITTI Odometry benchmark [32]. Stereo VO has the benefit over monocular VO that an initial estimate of the scene geometry, *i.e.* 3D points, can be computed from the 2D correspondences from the stereo image pair at each timeframe. Our stereo VO system finds an inlier subset of temporal $3D-2D$ correspondences and an approximate initial camera motion using AC-RANSAC [33]. After this step, the camera motion and 3D scene geometry are refined by minimizing the 2D reprojection error of the inlier point correspondences in the image domain.

A set of corresponding and non-corresponding image patches is considered from sequence 01 in the KITTI Odometry benchmark for the calibration. We consider a set of 70 frames, obtaining a total number of $364,704$ samples for each category, ensuring that the variability of stereo VO correspondences is sufficiently captured. Our corresponding image patches include matches between the left and right views of a stereo frame and also between consecutive frames, using a brute-force approach for matching the descriptors and a consistency check between the views. We carry out 10 runs per number-of-bits evaluation, averaging the normalized Kendall's tau estimates. The BDAE in Eq. 10 is evaluated for a certain number of bits in the range $K=\left\{1,\ldots,\binom{d}{2}\right\}$.

Fig. 5(a) depicts the BDAE measure. The BDAE is equal to 0.01 when $K^*=120$. As can be observed in Fig. 5(b), for $K=120$, the probability distributions of corresponding and non-corresponding image patches obtained by the LSH approximation are very similar compared to the ones obtained with the normalized Kendall's tau similarity. For this number of bits, the error in the approximation is very small and descriptor rankings remain nearly identical when adding more bits in the LSH approximation.

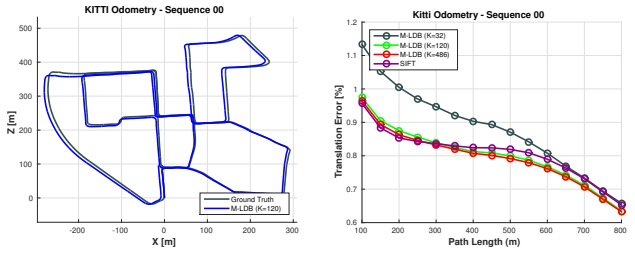
We use the sequence 00 as a test sequence for our calibration procedure. Stereo VO is performed considering two LSH approximations $(K=32,120)$ and the total length of the M-LDB descriptor $(K=486)$. In addition we show results considering the SIFT vector-based descriptor. We use the evaluation consistent with the KITTI benchmark and compute translational and rotational errors for all possible subsequences of length $(100,150,200,\ldots,800)$ meters.

Fig. 6(a) depicts the estimated trajectory considering the LSH approximation with $K^*=120$, showing that the estimated trajectory is very close to the ground truth trajectory.



(a) (b)

Fig. 5: ***Local binary descriptors calibration****. (a) BDAE for M-LDB. (b) Normalized Kendall's tau histogram and LSH approximation with* $K = 120$*.*



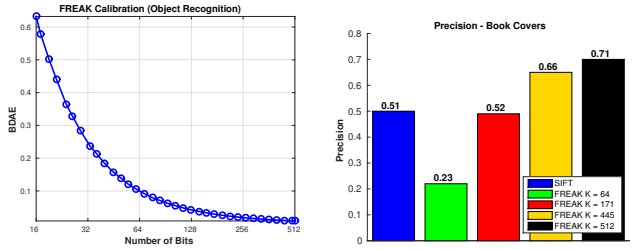
(a) (b)

Fig. 6: ***Visual odometry test sequence****. (a) Estimated trajectory w.r.t. ground truth (b) Average translation errors.*

Fig. 6(b) shows the average translation errors. We observe that the average translation errors are practically identical when using the LSH approximation with $K^* = 120$ compared to the total length of the binary descriptor $(K = 486)$ or the SIFT descriptor. When using $K = 32$ the translation error is slightly higher compared to the other values, since the relative approximation error as shown in Fig. 5(a) is still high and therefore descriptors are less discriminative, leading to a higher chance of introducing outliers in the set of correspondences. The rotational errors are very similar in all experiments, obtaining an average rotation error of $0.0029\ deg/m$. The calibration experiment shows that for a particular number of bits K^* in the LSH approximation, the relative approximation error is very small and we do obtain no further improvement in a particular image matching application by adding more bits. Therefore we can reduce descriptor computation time (extraction and matching), as well as storage requirements. For example, in our experiment the storage requirements when using $K^* = 120$ are reduced almost by a factor of 4 compared to the total length of the descriptor 486, and by a factor of 34 compared to SIFT $(128\ floats \times 32\ bits/float\ =\ 4096\ bits)$.

The proposed calibration method allows estimating the suitable number of bits K^* using just a small number of training images. It therefore saves computation time compared to exhaustively evaluating matching performance for different descriptor sizes on large sequences typical in visual odometry applications.

2) ***Object Recognition****:* We analyze the calibration of the FREAK descriptor for object recognition using the Stanford Mobile Visual Search dataset (SMVS) [34]. This dataset



(a) (b)

Fig. 7: ***Local binary descriptors calibration****. (a) BDAE for FREAK in the business cards category. (b) Precision scores for different values of* K*.*

contains 8 image categories (*book covers*, *business cards*, *CD covers*, *DVD covers*, *landmarks*, *paintings*, *video clips* and *text documents*). Each category has 100 reference images and 400 query images taken with different mobile phones and a Canon Powershot G11, with exception of the *landmarks* which contains 500 reference and query images.

The recognition process first finds keypoints and computes descriptors on the reference images. For each query image, we find matches between its keypoints and those of each reference image. We estimate the homography between the reference and query images using RANSAC with a reprojection error threshold of 2.5 pixels. The matches that do not conform to the homography are discarded. The reference image with more keypoint matches is selected if the number of matches is above a threshold set to 10, as in [34].

We evaluate binary descriptors performance by computing *precision* versus *number of bits* in the descriptor. Precision is computed as:

$$precision = \frac{\#correctly\ matched\ images}{\#images\ matched}. \qquad (12)$$

We use the first ten corresponding images from the *business cards* category to compute the BDAE. We considered an average number of $3,978$ samples per image pair. Then, we tested the performance of the FREAK descriptor considering all the images from the *book covers* category. Fig. 7(a) depicts the BDAE for the FREAK descriptor in the analyzed images from the *business cards* category. Note, that this scenario is more difficult than for the VO experiment. This is because the reference and query sets were taken with different cameras and under different image transformations such as blur, rotations and viewpoint changes. Fig. 7(b) shows the precision scores for the FREAK descriptor and SIFT in the *book covers* category. In this example, for a number of bits $K^* = 445$, the BDAE is equal to 0.01 and the precision is equal to 0.66, which corresponds to the 93% of the maximum precision value when using the maximum lenght of the descriptor.

VI. Conclusions and Future Work

In this paper we have introduced a novel perspective for the understanding and performance of pairwise test-based local binary descriptors. Under the understanding that local

binary descriptors can be interpreted as an LSH scheme that approximates Kendall's tau similarity between image patches, we showed that the Hamming distance between binary descriptors is an unbiased estimate of Kendall's tau similarity and derived expressions for its expectation and variance, explaining why descriptors rankings saturate at a certain number of bits in image matching applications. Our results reveal that excellent performance can be achieved using just a small fraction of the total number of bits from the whole descriptor, speeding up descriptor matching and reducing storage requirements considerably compared to vector-based descriptors. This insight is directly applicable to mobile devices and robotics applications with limited computational resources where the binary descriptor size may be adapted to different requirements. In addition, we have proposed a calibration method to automatically determine a suitable number of bits required in an image matching scenario. We hope that our findings will be useful for the design of efficient binary descriptors.

REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Intl. J. of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.

[3] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE features," in *Eur. Conf. on Computer Vision (ECCV)*, Firenze, Italy, 2012.

[4] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[5] M. Calonder, V. Lepetit, M. Özuysal, T. Trzcinski, C. Strecha, and P. Fua, "BRIEF: Computing a local binary descriptor very fast," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 34, no. 7, pp. 1281–1298, 2011.

[6] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Intl. Conf. on Computer Vision (ICCV)*, Barcelona, Spain, 2011.

[7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Intl. Conf. on Computer Vision (ICCV)*, Barcelona, Spain, 2011.

[8] X. Yang and K. T. Cheng, "Local Difference Binary for Ultrafast and Distinctive Feature Description," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 36, no. 1, pp. 188–194, 2014.

[9] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building Rome in a Day," in *Intl. Conf. on Computer Vision (ICCV)*, Kyoto, Japan, 2009.

[10] F. Perronin, Z. Akata, Z. Harchaoui, and C. Schmid, "Towards good practice in large-scale learning for image classification," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3482–3489.

[11] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A versatile and accurate monocular SLAM system," *IEEE Trans. Robotics*, 2015.

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

[13] R. Arroyo, P. F. Alcantarilla, L. M. Bergasa, and E. Romera, "Towards life-long visual localization using an efficient matching of binary sequences from images," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2015, pp. 6328–6335.

[14] J. Heinly, E. Dunn, and J. M. Frahm, "Comparative evaluation of binary features," in *Eur. Conf. on Computer Vision (ECCV)*, 2012, pp. 759–773.

[15] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 27, no. 10, pp. 1615–1630, 2005.

[16] D. Bekele, M. Teutsch, and T. Schuchert, "Evaluation of binary keypoint descriptors," in *Intl. Conf. on Image Processing (ICIP)*, 2013.

[17] A. Ziegler, E. Christiansen, D. Kriegman, and S. Belongie, "Locally uniform comparison image descriptor," in *Advances in Neural Information Processing Systems*, 2012.

[18] M. Kendall, "A new measure of rank correlation," *Biometrika*, vol. 30, pp. 81–89, 1938.

[19] T. Trzcinski, M. Christoudias, P. Fua, and V. Lepetit, "Boosting binary descriptors," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[20] P. F. Alcantarilla, J. Nuevo, and A. Bartoli, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," in *British Machine Vision Conf. (BMVC)*, Bristol, UK, 2013.

[21] K. Simonyan, A. Vedaldi, and A. Zisserman, "Learning local feature descriptors using convex optimisation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 36, no. 8, pp. 1573–1585, 2014.

[22] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua, "LDA-hash: Improved matching with smaller descriptors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 34, no. 1, pp. 66–78, 2012.

[23] J. Weickert and H. Scharr, "A scheme for coherence-enhancing diffusion filtering with optimized rotation invariance," *Journal of Visual Communication and Image Representation*, vol. 13, pp. 103–118, 2002.

[24] M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, 2002, pp. 380–388.

[25] J. Ji, S. Yan, J. Li, G. Gao, Q. Tian, and B. Zhang, "Batch-Orthogonal Locality-Sensitive Hashing for Angular Similarity," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 36, no. 10, pp. 1963–1974, 2014.

[26] R. Kumar and S. Vassilvitskii, "Generalized distances between rankings," in *Proceedings of the 19th International Conference on World Wide Web (WWW)*, 2010, pp. 571–580.

[27] K. Pal and S. Michel, "An LSH Index for Computing Kendall's Tau over Top-k Lists," in *Proceedings of the 17th International Workshop on the Web and Databases (WebDB)*, 2014.

[28] A. Andoni and P. Indyk, "Near-optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions," *Commun. ACM*, vol. 51, no. 1, pp. 117–122, 2008.

[29] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from Internet photo collections," *Intl. J. of Computer Vision*, vol. 80, no. 2, pp. 189–210, 2008.

[30] C. Strecha, W. von Hassen, L. V. Gool, P. Fua, and U. Thoennessen, "On benchmarking camera calibration and multi-view stereo for high resolution imagery," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, Alaska, USA, 2008.

[31] M. Brown, G. Hua, and S. Winder, "Discriminative learning of local image descriptors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 33, no. 1, pp. 43–57, 2011.

[32] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Intl. J. of Robotics Research*, 2013.

[33] P. Moulon, P. Monasse, and R. Marlet, "Adaptive structure from motion with a contrario model estimation," in *Asian Conf. on Computer Vision (ACCV)*, 2012, pp. 257–270.

[34] V. R. Chandrasekhar, D. M. Chen, S. S. Tsai, N.-M. Cheung, H. Chen, G. Takacs, Y. Reznik, R. Vedantham, R. Grzeszczuk, J. Bach, and B. Girod, "The Stanford mobile visual search data set," in *Proceedings of the second annual ACM conference on Multimedia systems*, 2011.