Quantized Embeddings of Scale-Invariant Image Features for Mobile Augmented Reality

Mu Li[1], Shantanu Rane[2] and Petros Boufounos[2]

[1] *Pennsylvania State University, University Park, PA, USA.*
[1] mql5192@psu.edu

[2] *Mitsubishi Electric Research Laboratories, Cambridge, MA, USA.*
[2] {rane, petrosb}@merl.com

***Abstract*—Randomized embeddings of scale-invariant image features are proposed for retrieval of object-specific meta data in an augmented reality application. The method extracts scale invariant features from a query image, computes a small number of quantized random projections of these features, and sends them to a database server. The server performs a nearest neighbor search in the space of the random projections and returns meta-data corresponding to the query image. Prior work has shown that binary embeddings of image features enable efficient image retrieval. This paper generalizes the prior art by characterizing the tradeoff between the number of random projections and the number of bits used to represent each projection. The theoretical results suggest a bit allocation scheme under a total bit rate constraint: It is often advisable to spend bits on a small number of finely quantized random measurements rather than on a large number of coarsely quantized random measurements. This theoretical result is corroborated via experimental study of the above tradeoff using the ZuBuD database. The proposed scheme achieves a retrieval accuracy up to 94% while requiring the mobile device to transmit only 2.5 kB to the database server, a significant improvement over 1-bit quantization schemes reported in prior art.**

I. Introduction

Augmented Reality is one of the most significant applications to leverage the recent advances in mobile device technology. In addition to using smartphones and tablet computers to sense the real world by capturing images, videos and sounds, people can augment that experience by overlaying useful information on the real world data. A well-known example of augmented reality is Google Goggles, an application that allows a user to obtain meta information about her environment, such as overlaying the name of a historical landmark onto a recently acquired photograph, or recovering information about a consumer product using an image of the product's barcode. To make such applications feasible and powerful, it is necessary to exploit recent advances in image recognition while recognizing the limitations on the speed, power consumption, memory, processing time, and communication bandwidth at the mobile device. Thus, in a typical augmented reality application, a mobile device must efficiently transmit the salient features of the image that it has captured to a remote database that contains a large number of images. The database server should quickly determine whether the query matches an entry in the database and return suitable augmented information to the mobile device.

Much of the success of image-based augmented reality applications is owed to the development of image descriptors such as SIFT [1], SURF [2], GIST [3] and allied techniques. Of these, GIST captures global properties of the image and has been used for image matching. SIFT and SURF capture local details at several salient points in an image, and therefore, have been used to match local features or patches. They can also be used for image matching and retrieval by combining hypotheses from several patches using, for example, the popular Bag-of-Features approach [4]. A comparative study of image descriptors has shown that Scale Invariant Feature Transformation (SIFT) features have the highest robustness against common image deformations such as translation, rotation, and a limited amount of scaling [5]. Nominally, a SIFT feature vector for a single salient point in an image is a real-valued, unit-norm 128-dimensional vector. This results in a prohibitively large bit rate required to transmit the SIFT features to a database server for the purpose of matching, especially if features from several salient points are needed for reliable matching.

There is a large body of work on training-based methods to compress image descriptors [6]–[10]. Boosting Similarity Sensitive Coding (BoostSSC) and Restricted Boltzmann Machines (RBM) have been proposed for learning compact GIST codes for content-based image retrieval [6]. Alternatively, semantic hashing can be transformed into a spectral hashing problem in which it is only necessary to calculate eigenfunctions of the GIST features, providing better retrieval performance than BoostSSC and RBM [7]. Besides these relatively recently developed machine learning algorithms, some classical training-based techniques such as Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) have also been used to generate compact image descriptors. In particular, PCA has been used to produce small image descriptors by applying techniques such as product quantization [8] and distributed source coding [9]. Alternatively, small image descriptors were obtained by applying LDA to SIFT-like descriptors followed by binary quantization [10].

While training-based methods perform accurately in traditional image retrieval, they may become cumbersome in augmented reality applications, where the database can keep growing as new landmarks, products, etc. are added, resulting in new image statistics and necessitating repeated training. As a source coding-based alternative to training-based dimensionality reduction, a low-bitrate descriptor has been constructed using a Compressed Histogram of Gradients (CHoG) specifically for augmented reality applications [11]. In this method, gradient distributions are explicitly compressed, resulting in low-rate scale invariant descriptors. Two other techniques have been proposed for efficient remote image matching based on Locality Sensitive Hashing (LSH, [12]), which is computationally simpler, but less bandwidth-efficient than CHoG, and does not need training. In the first, random projections are computed from scale invariant features followed by one-bit quantization, and the resulting descriptors are used to establish visual correspondences between images captured in a wireless camera network [13]. In the second, the same technique is applied to content-based image retrieval, and a bound is obtained for the minimum number of bits needed for a specified accuracy of nearest neighbor search [14]. However, these works do not consider the tradeoff between dimensionality reduction and quantization. It is

The present work was carried out when M. Li was an intern at Mitsubishi Electric Research Laboratories, Cambridge, MA.

the intention of the present work to quantify this tradeoff, and to show both theoretically and experimentally that finer quantization with fewer random projections can be more bandwidth-efficient than LSH, while retaining the advantages of simplicity and low complexity.

The remainder of this paper is organized as follows: Section II provides a theoretical justification for performing a nearest neighbor search using random projections, rather than using the original image features. Starting from the Johnson-Lindenstrauss Lemma, we describe a tradeoff between the number of random projections and the fidelity of representing each projection, i.e., the quantization step size. Section III leverages this result in a simple augmented reality application in which a server returns meta data corresponding to the query image, based on an approximate k-nearest neighbor search in the space of quantized embeddings of scale-invariant image features. Section IV describes experimental results of simulating the augmented reality application using SIFT features extracted from images from the ZuBuD database. Section V concludes the paper.

II. Quantized Randomized Embeddings

Our work relies on a low-dimensional embedding of scale-invariant feature points extracted from images. The use of embeddings is justified by the following result which forms the starting point of our theoretical development.

Theorem 1 *(Johnson-Lindenstrauss Lemma [15]) For a real number $\epsilon \in (0,1)$ let there be a positive integer k such that*

$$k \geq \frac{4}{\epsilon^2/2 - \epsilon^3/3} \ln n$$

Then, for any set $\mathcal{X} \subset \mathbb{R}^d$ that contains n points, there is a mapping $f : \mathbb{R}^d \to \mathbb{R}^k$, computable in randomized polynomial time, such that for all $\mathbf{u}, \mathbf{v} \in \mathcal{X}$,

$$(1-\epsilon)\|\mathbf{u}-\mathbf{v}\|^2 \leq \|f(\mathbf{u}) - f(\mathbf{v})\|^2 \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\|^2$$

In the above result and in the following development, $\|\cdot\|$ represents the ℓ_2 norm. A key point is that for a given ϵ, the dimensionality k of the points in the range of f is independent of the dimensionality of points in $\mathcal{X}$ and proportional to the logarithm of number of points in $\mathcal{X}$. Since k grows like $\ln n$, the Johnson-Lindenstrauss Lemma establishes a dimensionality reduction result, in which any set of n points in d-dimensional Euclidean space can be embedded into k-dimensional Euclidean space. This is extremely beneficial for querying huge databases (i.e., large n) with several attributes (i.e., large d). One way to construct the embedding function f is to project the points from $\mathcal{X}$ onto a spherically random hyperplane passing through the origin. In practice, this is accomplished by multiplying the data vector with a matrix whose entries are drawn from a specified distribution. In particular, a matrix with iid. $\mathcal{N}(0,1)$ entries provides the distance-preserving properties in Theorem 1 with high probability. The following result, due to [16], [17], makes this notion precise.

Theorem 2 *For real numbers $\epsilon, \beta > 0$, let there be a positive integer k such that*

$$k \geq \frac{4 + 2\beta}{\epsilon^2/2 - \epsilon^3/3} \ln n \tag{1}$$

Consider a matrix $\mathbf{A} \subset \mathbb{R}^{k \times d}$, whose entries $a(i,j)$ are drawn iid. from a $\mathcal{N}(0,1)$ distribution. Let there be a set $\mathcal{X} \subset \mathbb{R}^d$ that contains n points. Then, for all $\mathbf{u} \in \mathcal{X}$, the mapping $f(\mathbf{u}) = \frac{1}{\sqrt{k}}\mathbf{A}\mathbf{u}$ satisfies the distance preserving property in Theorem 1 with probability at least as large as $1 - n^{-\beta}$.

By construction, $f(\mathbf{u})$ is a k-dimensional embedding of a d-dimensional vector. Theorem 2 holds for other distributions on $a(i,j)$ besides the normal distribution [18]. In what follows, however, we consider only the normal gaussian case. We are especially interested in the distance-preserving property for *quantized* embeddings, i.e., the case when a uniform scalar quantizer is applied independently to each element of $f(\mathbf{u})$ and $f(\mathbf{v})$. Theorem 2 says that, in the unquantized case, the embedding f is ϵ-accurate with probability $1 - n^{-\beta}$. The question we ask is: What happens to the embedding accuracy when quantization is employed to reduce the bit rate required to store or transmit the embeddings? Furthermore, what is the tradeoff between quantization and the number of projections k that can be transmitted while remaining below a specified bit budget? The following proposition is the first step in understanding those tradeoffs.

Proposition 1 *For real numbers $\beta > 0$ and $\epsilon \in (0,1)$, let there be a positive integer k that satisfies (1). Consider a matrix $\mathbf{A} \subset \mathbb{R}^{k \times d}$, whose entries $a(i,j)$ are drawn iid. from a $\mathcal{N}(0,1)$ distribution. Let there be a set $\mathcal{X} \subset \mathbb{R}^d$ that contains n points. For any vector $\mathbf{w}$, let $q(\mathbf{w})$ be an uniform scalar quantizer with step size Δ applied independently to each element of $\mathbf{w}$. Then, for all $\mathbf{u}, \mathbf{v} \in \mathcal{X}$, the mapping $g(\mathbf{u}) = \frac{1}{\sqrt{k}} q(\mathbf{A}\mathbf{u})$ satisfies*

$$(1-\epsilon)\|\mathbf{u}-\mathbf{v}\| - \Delta \leq \|g(\mathbf{u}) - g(\mathbf{v})\| \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\| + \Delta$$

with probability at least as large as $1 - n^{-\beta}$.

Proof: First, note that quantization, $q(\mathbf{A}\mathbf{u})$, introduces error at most $\Delta/2$ per dimension, i.e., $\|\mathbf{A}\mathbf{u} - q(\mathbf{A}\mathbf{u})\| \leq \sqrt{k}\Delta/2$ for any $\mathbf{x}$. Using this, along with $f(\mathbf{u}) = \frac{1}{\sqrt{k}}\mathbf{A}\mathbf{u}$ as in Theorem 2, we get $\|f(\mathbf{u}) - g(\mathbf{u})\| \leq \Delta/2$. Now, take the square roots in the statement of Theorem 1 noting that, for $\epsilon \in (0,1)$, $1 + \epsilon \geq \sqrt{(1+\epsilon)}$ and $1 - \epsilon \leq \sqrt{(1-\epsilon)}$, we get

$$(1-\epsilon)\|\mathbf{u}-\mathbf{v}\| \leq \|f(\mathbf{u}) - f(\mathbf{v})\| \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\|$$

Then, the right half of the theorem statement follows from the triangle inequality as

$$\begin{aligned}\|g(\mathbf{u}) - g(\mathbf{v})\| &\leq \|g(\mathbf{u}) - f(\mathbf{u})\| + \|f(\mathbf{u}) - f(\mathbf{v})\| + \|f(\mathbf{v}) - g(\mathbf{v})\| \\ &\leq \Delta/2 + (1+\epsilon)\|\mathbf{u}-\mathbf{v}\| + \Delta/2.\end{aligned}$$

The left half is proved similarly. ■

It is evident from the proposition that the scalar quantization interval Δ is critical to the accuracy of the quantized embedding. This, in turn, depends on the design of the scalar quantizer and the bit-rate B used to encode each coefficient. In this work, we consider a finite uniform scalar quantizer with saturation levels $\pm S$, that we assume to be set such that saturation is sufficiently rare and can be ignored. Thus, B bits are used to uniformly divide the range of the quantizer, $2S$, making the quantization interval $\Delta = 2^{-B+1}S$. Using R to denote the total rate available to transmit the k measurements, i.e., setting $B = R/k$ bits per measurement, the quantization interval is $\Delta = 2^{-R/k+1}S$. Thus the tradeoff, implicit in Proposition 1, between number of measurements and number of bits per measurement becomes more explicit:

$$\begin{aligned}(1-\epsilon)\|\mathbf{u}-\mathbf{v}\| - 2^{-\frac{R}{k}+1}S& \\ \leq \|g(\mathbf{u}) - g(\mathbf{v})\| &\leq \\ (1+\epsilon)\|\mathbf{u}-\mathbf{v}\| &+ 2^{-\frac{R}{k}+1}S,\end{aligned} \tag{2}$$

By shifting the origin to $\mathbf{v}$, we can tighten the bound in the statement to: $(1-\epsilon)\|\mathbf{u}-\mathbf{v}\| - \frac{\Delta}{2} \leq \|g(\mathbf{u}) - g(\mathbf{v})\| \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\| + \frac{\Delta}{2}$

Specifically, increasing the number of measurements for a fixed rate R, decreases the available rate per measurement and, therefore, increases the quantization interval Δ. This, in turn, increases the quantization error ambiguity, given by the additive factor $\pm 2^{-\frac{R}{k}+1}S$. Furthermore, increasing the number of measurements reduces ϵ and, therefore, reduces the ambiguity due to Theorem 2, given by the multiplicative factor $(1 \pm \epsilon)$. Note that, for fixed β and n, ϵ scales approximately proportionally to $1/\sqrt{k}$ when small.

There are two issues we do not address in the development above: non-uniform quantization and saturation. A non-uniform scalar quantizer, tuned to the distribution of the measurements, may improve embedding performance. However, it will still suffer the same tradeoff between number of bits per measurement and number of measurements. Detailed theoretical analysis of non-uniform quantizers is beyond the scope of this paper. Similarly, adjusting the saturation rate of the quantizer is a way to tune the quantizer to the distribution of the measurements. Reducing the range of the quantizer, S, reduces the quantization interval Δ and the ambiguity due to quantization. However, it increases the probability of saturation and, consequently, the unbounded error due to saturation, making the above model invalid and the theoretical bounds inapplicable. For compressive sensing reconstruction from quantized random projections, careful tuning of the saturation rate has been shown to improve performance [19]. However, taking quantization appropriately into account in the context of nearest-neighbor computation and Johnson-Lindenstrauss embeddings is not as straightforward and we do not attempt it in this paper.

We also note that the above theoretical development partially breaks down for quantization at 1-bit per measurement which is performed just by keeping the sign of the projection. If one signal in the set of interest is a positive scalar multiple of another, then the two signals will be indistinguishable in the embedding. While the guarantees still hold for bounded norm signals, they are often too loose to be useful. Tighter bounds can instead be developed if we are interested in the angles between two signals—i.e., their correlation—instead of their distance. Examples of such developments for 1-bit quantization can be found in [20]–[22].

III. Embeddings of Scale-Invariant Image Features

We now describe the proposed framework for retrieving object-specific metadata from a query image using quantized embeddings of scale-invariant features. The application scenario is as follows: Alice wants more information about a query object, such as history of a monument, or nutrition information for a food item. She uses a mobile device – usually a tablet computer or a smartphone – to acquire an image of the query object. The device sends information about the query object to a database server. The server locates the object in the database that most closely matches the query signal according to some predetermined distance criterion, and transmits the meta-data of that object back to Alice's mobile device.

For such a scheme to be practical, the following requirements must be met: (1) The mobile device should be able to generate a query signal at low complexity (2) The bandwidth required to transmit the query signal to the database server must be small (3) The server must have either a fast algorithm or sufficient computing power to quickly process the meta-data request (4) The server must transmit the meta-data efficiently to the mobile device. In this work, we are concerned with the first two requirements, which are the most challenging. We note that most methods that speed up server-based matching and enhance meta-data compression will only supplement the advantages of the proposed system. Algorithm 1 describes the steps taken by the server to prepare its database before deploying the augmented reality service.

Algorithm 1 Database Preparation At Server

```
Initialize the random projection matrix $\mathbf{A} \in \mathbb{R}^{k \times d}$, where the elements $a(i,j) \sim \mathcal{N}(0,1)$.
Acquire images $\mathbf{J}_1, \mathbf{J}_2 ..., \mathbf{J}_t$ of $s$ objects, where $s \le t$. Create meta-data $\mathbf{D}_i, i \in \{1,2,...,s\}$ for each object.
Run a scale-invariant feature extraction algorithm on each image $\mathbf{J}_i, i \in \{1,2,...,t\}$ which will return several $d$-dimensional features from each image. The number of features obtained from each image need not be equal. Using all the feature vectors thus obtained, construct the matrix $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_N]$ which contains feature vectors from *all images of all objects* in the database. Typically, $N \gg s$.
Compute the matrix $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N] = \mathbf{AV} \in \mathbb{R}^{k \times N}$, where each $\mathbf{w}_i$ is a $k$-dimensional random projection of the corresponding $\mathbf{v}_i$.
Store a lookup vector $\mathbf{\Lambda} \subset \{1,2,...,s\}^N$ where the element $\lambda(i), i \in \{1,2,...,N\}$ indexes the object from which the vector $\mathbf{w}_i$ was extracted.
```

Next, we describe the querying procedure in Algorithm 2, which is executed at the mobile device using the same random projection matrix $\mathbf{A}$ as the server. The distribution of the $a(i,j)$ can be approximated by a pseudorandom number generator. It is assumed that, the seed of the pseudorandom number generator is sent to the mobile device as a one-time software update or included as part of the client software installation. This seed ensures that the mobile device and the server generate the same realization of $\mathbf{A}$.

Algorithm 2 Query Procedure At Mobile Device

```
Initialize the random projection matrix $\mathbf{A} \in \mathbb{R}^{k \times d}$, where the elements $a(i,j) \sim \mathcal{N}(0,1)$.
Acquire query image $\mathbf{I}$.
Run the scale-invariant feature extraction algorithm on $\mathbf{I}$ to derive the matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_M]$, where $\mathbf{x}_i$ is a $d$-dimensional feature vector corresponding to the $i^{\text{th}}$ key point descriptor from the image $\mathbf{I}$.
Compute the matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_M] = \mathbf{AX} \in \mathbb{R}^{k \times M}$, where each $\mathbf{y}_i$ is a $k$-dimensional random projection of the corresponding $\mathbf{x}_i$.
Compute the matrix of quantized random projections $\mathbf{Q} = q(\mathbf{Y})$, where the function $q(\cdot)$ is a scalar quantizer that takes each of the $kM$ elements of $\mathbf{Y}$ and produces, for each element, an integer quantization index in the set $\{0,1,...,L-1\}$.
Transmit the matrix $\mathbf{Q}$ to the server, using element-wise fixed length coding of the quantization indices. Thus, each quantization index is represented by $\lceil \log_2 L \rceil$ bits.
```

Based on Algorithm 2, the computational complexity at the mobile device is primarily determined by the scale-invariant feature extraction algorithm, and one matrix multiplication. The number of bits transmitted by the mobile device to the server is thus $kM\lceil \log_2 L \rceil$ bits. To save on transmit power, the mobile device may reduce the number of random projections k, the quantization levels L, or the number of feature vectors M extracted from the query image.

Next, Algorithm 3 describes the approximate nearest neighbor procedure executed by the server. Briefly, nearest neighbors are found in the space of the quantized embeddings of image descriptors. The

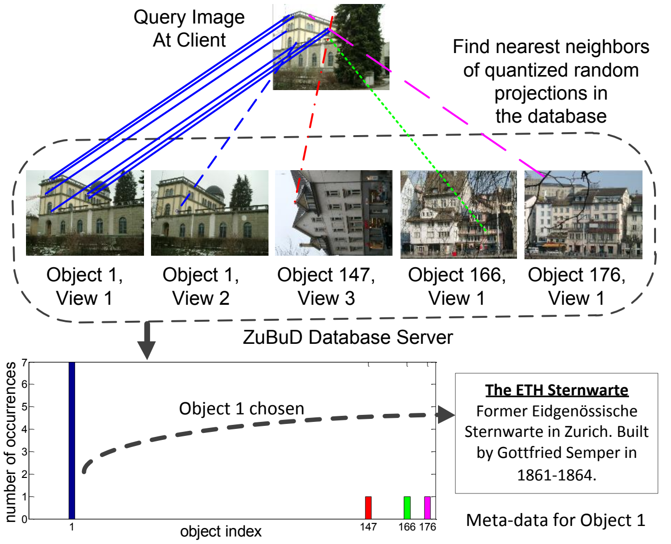

Fig. 1. The server uses Algorithm 3 to find the image that matches the query image using approximate nearest neighbor search in the space of the quantized embeddings of scale invariant features .

nearest neighbors are then aggregated to obtain the matching image, and thence its meta-data. This is pictorially depicted in Fig. 1. In our implementation of Algorithm 3, we use $r = 10$.

Algorithm 3 Approximate Nearest Neighbors

```
1: Initialize an $s$-dimensional “histogram” vector $\mathbf{h}$ to all zeros.
2: Receive $\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_M]$ as a query. Receive $L$, the number of quantization levels.
3: Invert the quantization function $q(\cdot)$ and obtain the reconstructed random projection matrix $\hat{\mathbf{Y}} = q^{-1}(\mathbf{Q}) = [\hat{\mathbf{y}}_1, \hat{\mathbf{y}}_2, ..., \hat{\mathbf{y}}_M]$.
4: Compute $\hat{\mathbf{W}} = q^{-1}(q(\mathbf{W}))$, which contains the quantized reconstruction of all $k$-dimensional random projections corresponding to all $t$ images of $s$ objects.
5: for each $i \in \{1, 2, ..., M\}$ do
6:     Find the nearest neighbor of $\hat{\mathbf{y}}_i$ among $\hat{\mathbf{w}}_1, \hat{\mathbf{w}}_2, ..., \hat{\mathbf{w}}_N$.
7: Out of these $M$ nearest neighbor pairs, and choose $r$ pairs $(\hat{\mathbf{y}}_{(j)}, \hat{\mathbf{w}}_{(j)}), j = 1, 2, ..., r$ that are closest in Euclidean distance.
8: for each $\hat{\mathbf{w}}_{(j)}, j = 1, 2, ..., r$ do
9:     Read off the index $\alpha_j \in \{1, 2, ..., s\}$ of the object from which the element occurs. This is readily available from the lookup table $\mathbf{\Lambda}$ from Algorithm 1.
10:    Increment $h(\alpha_j)$ by 1.
11: Set the nearest object to the query image as $\arg\max_{\alpha} h(\alpha)$ and send the meta-data of this object back to the querying device.
```

If the quantizer codebook is known beforehand, the server can optionally precompute $\hat{\mathbf{W}}$ in the database preparation stage in Algorithm 1, rather than repeating Step 4 of Algorithm 3 for each query image. Finally, we note that if more images, or superior quality images or richer meta-data become available at a later time, they can be easily appended to the database without affecting the querying algorithm executed by the mobile device.

IV. Experiments

We conducted experiments on a public database to evaluate the performance of meta-data retrieval using quantized embeddings of scale-invariant features, to compare it against LSH-based techniques [13], [14] and to validate our theoretical analysis. We used the ZuBuD database [23], which contains 1005 images of 201 buildings in the city of Zurich. There are 5 images of each building taken from different viewpoints. The images were all of size 640 × 480 pixels, compressed in PNG format. One out of the 5 viewpoints of each building was randomly selected as the query image, forming a query image set of $s = 201$ images. The server's database then contains the remaining 4 images of each building, for a total of $t = 804$ images.

For the scale-invariant feature space used in Step 3 of Algorithms 1 and 2, we use the popular SIFT feature space [1]. Thus, SIFT features are extracted from the server's images to construct the matrix $\mathbf{V}$ in Algorithm 1 and the matrix $\mathbf{X}$ in Algorithm 2. Clearly, it is desirable that the meta-data transmitted to the mobile device corresponds to the correct image, i.e., the nearest neighbor of the query image. To measure the fidelity of the algorithm, we define the performance metric as follows: Let N_q be the number of query images. In our experiment $N_q = 201$. Let N_c be the number of query images for which the meta-data transmitted at the end of Algorithm 3 corresponds to the correct image, as verified against ground truth. Then, define $P_{cor} = \mathrm{E}(N_c/N_q)$ where the expectation is taken over the randomness in the experiment, namely the realization of the random projection matrix $\mathbf{A}$. We repeat each experiment 30 times, using a different random realization of $\mathbf{A}$ each time, average the N_c/N_q values, and report the mean value as P_{cor}.

A. Performance Comparison with LSH-based schemes

Both the LSH-based schemes [13], [14] use random projections of the SIFT vectors followed by 1-bit quantization according to the sign of the random projections. We compared the accuracy of meta-data retrieval achieved by the LSH-based schemes with our multi-bit quantization approach. Fig. 2 shows the variation of P_{cor} against the number of projections for the LSH-based schemes (in blue). This is significantly inferior to meta-data retrieval based on unquantized projections. Between the two extremes lie the performance curves of the multibit quantization schemes. Using 4 or 5 bits per dimension nearly achieves the performance of unquantized random projections.

Next, we determine experimentally whether there is a quantizer that is "optimal" in the sense of achieving the highest P_{cor} per bit when the total bit budget allocated for transmitting all the quantized random projections was fixed. To investigate this, we plotted P_{cor} against the number of bits needed to transmit each k-dimensional descriptor, as shown in Fig. 3. When an L-level scalar uniform quantizer is used independently on each dimension of the descriptor, the number on the horizontal axis is simply $k\lceil\log_2 L\rceil$ bits. This facilitates a comparison of all schemes for the same number of transmitted bits. The multibit quantizer again gives higher probability of correct retrieval than the LSH-based schemes, confirming that taking few finely quantized projections can outperform taking many coarsely quantized projections. Particularly, the 3 and 4-bit quantizers provide the highest P_{cor} for a given bit budget, outperforming the 5-bit quantizer and indicating the existence of an optimal quantizer.

Each element of a SIFT vector lies between 0 and 1, and SIFT vectors have approximately unit norm. We compared the partitions generated by applying the Lloyd algorithm [24] to random projections of SIFT vectors with those generated by a uniform quantizer. Owing to the small dynamic range, as the number of bits per dimension is increased, the partitions from Lloyd quantization rapidly coincide with those for a uniform quantizer applied to a $\mathcal{N}(0, 1)$ random variable. Consequently the performance of uniform quantization and Lloyd quantization was observed to be nearly identical.

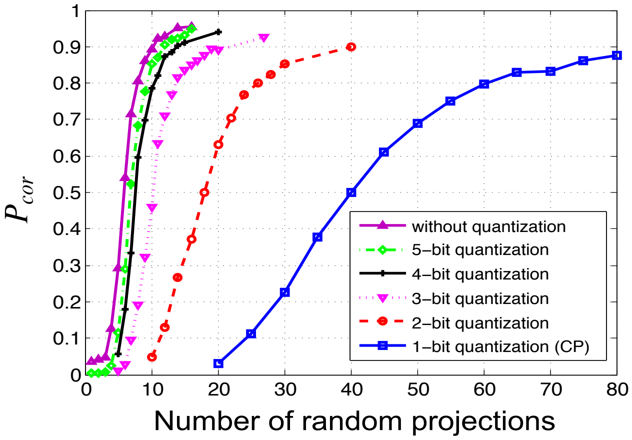

Fig. 2. Multi-bit quantization with fewer random projections outperforms LSH-based schemes [13], [14] which employ 1-bit quantization with a large number of random projections.

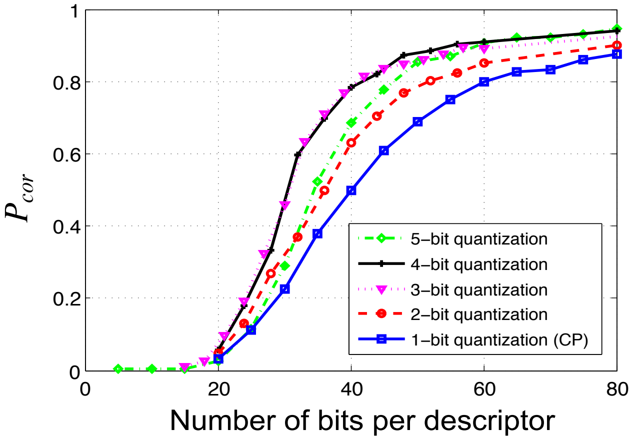

Fig. 3. When the bit budget allocated to each descriptor (vector) is fixed, the best retrieval performance is achieved with 3-bit and 4-bit quantization.

B. Experimental Validation of Theoretical Result

From (2), the error between the ℓ_2 distance between the SIFT vectors and the ℓ_2 distance between their quantized embeddings is $\epsilon\|\mathbf{u}-\mathbf{v}\| + 2^{-\frac{R}{k}+1}S$ with overwhelming probability. We now plot this error value against the number of bits per dimension R/k for the data in our experiments. To do this, we count the total number of SIFT vectors used, $n = 278345$. The average value of the pairwise distances between SIFT vectors, i.e., the average value of $\|\mathbf{u}-\mathbf{v}\|$, was computed to be 1.026. Further, the database server has unquantized versions of all random projections from which we obtain the maximum saturation level $S = 5.203$. Lastly, for Proposition 1 to hold with very high probability, say 0.9999, we must choose $\beta = -(\log 10^{-4})(\log n)^{-1} = 0.7347$. Plugging in the values of β and n in (1) and neglecting the ϵ^3 term in (1), we get $\epsilon \leq \sqrt{((1/k)(8+4\beta)\ln n)} = 11.71/\sqrt{k}$.

Using the average value for $\|\mathbf{u}-\mathbf{v}\|$ and the worst case value for ϵ, we plot the different embedding error in Fig. 5 for various values of the total bit budget R. It is observed that, over a large range of R values, there exists an “optimum” value of bits allocated per random projection, equivalently an optimum quantizer that achieves the lowest embedding error. This corroborates the experimental finding in the previous subsection.

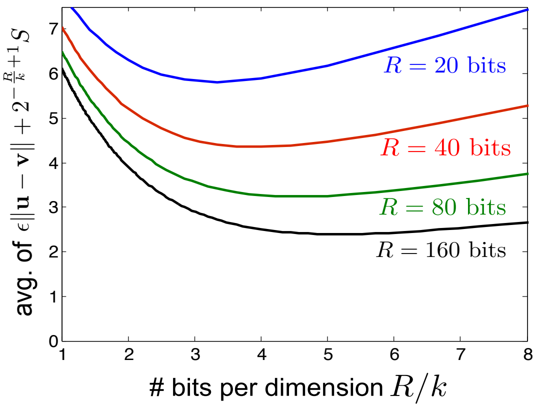

Fig. 4. For the database used, the derived theoretical result (2) indictates that, over a large range of the bit budget R, there is an optimal bit allocation that achieves the lowest embedding error. This is consistent with the experimental results obtained without recourse to the theory in Fig. 3.

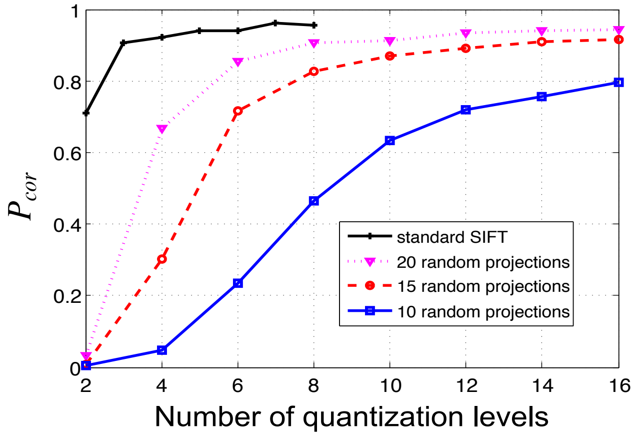

Fig. 5. Because of their low dimensionality, quantized random embeddings achieve excellent retrieval performance at a fraction of the bit rate required by a scheme that quantizes the underlying SIFT vectors.

C. Influence of Codebook Size on Retrieval Performance

To investigate the influence of quantizer granularity on retrieval performance, we plot the probability of retrieving the correct metadata against a linearly increasing codebook size in Fig. 5. As a benchmark, we also plot the P_{cor} values for quantization applied directly to the 128-dimensional SIFT features ($d = 128$). As expected, for a given number of quantization levels, P_{cor} increases when the number of random projections is increased. Further, for a fixed number of random projections, P_{cor} increases when the number of quantization levels – equivalently, the bits per dimension – is increased. The quantized embeddings are significantly more bit-rate efficient than SIFT. For instance, in order to achieve $P_{cor} = 0.9$, SIFT requires at least $128 \times \lceil \log_2 3 \rceil = 256$ bits per vector using the simple constant-length encoding scheme. In comparison, the quantized embeddings require only about 60 bits by using either (a) 15 projections with 4 bits per dimension, or (b) 20 projections with 3 bits per dimension.

D. Influence of the Query Size on Retrieval Performance

The query size is the total number of bits transmitted by the mobile device to the database server. This is the product of the number of random projections k, the bits per dimension R/k, and the total

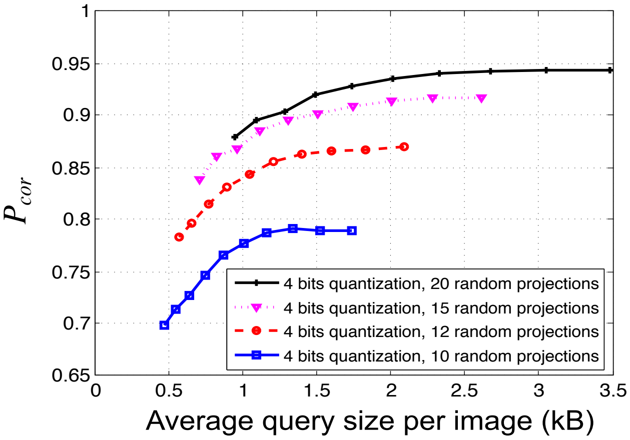

Fig. 6. Proposed scheme with 20 random projections & 4-bit quantization achieves 94 % accuracy with a small average query size of 2.5 kB per image.

number of feature vectors M extracted from the query image in Algorithm 2. Clearly, a lower query size is desirable. To change the number of extracted SIFT features M, we change the threshold below which the maxima from the Difference-Of-Gaussian (DOG) scale space are ignored. In practice, the number of features extracted for a preset threshold is image-dependent, thus the query size is also image-dependent. Therefore, we report results based on the average query size for 201 images from our query set in Fig. 6.

It is observed that highly accurate meta-data retrieval with $P_{cor} = 0.94$ is achieved with average query size 2.5 kilobytes per query image. This is much smaller than the bit rate needed to transmit the JPEG compressed query image using quality factor 80, which requires 58.5 kilobytes per image averaged over the query set. This justifies our proposal to transmit quantized embeddings of SIFT features rather than sending the JPEG compressed image to the database server. Recall that we use a very simple method, i.e., constant length coding, to transmit the quantized random projections. If a slight increase in encoder complexity is tolerated, an entropy coder applied to the quantization indices will further improve the performance.

V. Discussion

This work has shown that randomized embeddings of scale invariant image features can be used for image retrieval while consuming much lower bit rate compared with directly using the scale invariant features. The derived result and our experiments suggest that when very few bits are available, then it makes sense to allocate them towards increasing the number of incoherent measurements. However, after a certain minimum number of random measurements is satisfied, it is more beneficial to utilize any additional bits toward representing these projections with high fidelity rather than continuing to increase the number of coarsely quantized random projections.

There are several interesting avenues for future work in this area. One area of interest is to study the privacy benefits of using randomized embeddings. Specifically, is it possible to construct an embedding that reveals the true distance between two images (or image features) only if they are close, but reveals no information if the images are far apart? In this paper, we compared our approach against other randomized embedding approaches proposed in [13], [14]. An additional item of interest is to compare the retrieval performance vis-a-vis encoding complexity and upload bandwidth of our scheme with methods such as [11] that explicitly compress the scale-invariant descriptors using a vector quantizer.

References

[1] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

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

[3] A. Oliva and A. Torralba, “Modeling the shape of the scene: A holistic representation of the spatial envelope,” *International Journal of Computer Vision*, vol. 42, pp. 145–175, 2001.

[4] J. Sivic and A. Zisserman, “Efficient visual search of videos cast as text retrieval,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 4, pp. 591–606, Apr. 2009.

[5] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615 –1630, Oct. 2005.

[6] A. Torralba, R. Fergus, and Y. Weiss, “Small codes and large image databases for recognition,” in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, Jun. 2008, pp. 1 –8.

[7] Y. Weiss, A. Torralba, and R. Fergus, “Spectral hashing,” in *Advances in Neural Information Processing Systems 21*, D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou, Eds., 2009, pp. 1753–1760.

[8] H. Jegou, F. Perronnin, M. Douze, J. Sanchez, P. Perez, and C. Schmid, “Aggregating local images descriptors into compact codes,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, p. 1, 2011.

[9] C. Yeo, P. Ahammad, and K. Ramchandran, “Coding of image feature descriptors for distributed rate-efficient visual correspondences,” *International Journal of Computer Vision*, vol. 94, pp. 267–281, 2011, 10.1007/s11263-011-0427-1.

[10] C. Strecha, A. Bronstein, M. Bronstein, and P. Fua, “LDAHash: Improved matching with smaller descriptors,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 1, pp. 66 –78, Jan. 2012.

[11] V. Chandrasekhar, G. Takacs, D. Chen, S. Tsai, Y. Reznik, R. Grzeszczuk, and B. Girod, “Compressed histogram of gradients: A low-bitrate descriptor,” *International Journal of Computer Vision*, vol. 96, pp. 384–399, 2012.

[12] 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, Jan. 2008.

[13] C. Yeo, P. Ahammad, and K. Ramchandran, “Rate-efficient visual correspondences using random projections,” in *IEEE International Conference on Image Processing*, Oct. 2008, pp. 217 –220.

[14] K. Min, L. Yang, J. Wright, L. Wu, X.-S. Hua, and Y. Ma, “Compact projection: Simple and efficient near neighbor search with practical memory requirements,” in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, Jun. 2010, pp. 3477 –3484.

[15] W. Johnson and J. Lindenstrauss, “Extensions of Lipschitz mappings into a Hilbert space,” *Contemporary Mathematics*, vol. 26, pp. 189 – 206, 1984.

[16] S. Dasgupta and A. Gupta, “An elementary proof of a theorem of Johnson and Lindenstrauss,” *Random Structures & Algorithms*, vol. 22, no. 1, pp. 60–65, 2003.

[17] P. Indyk and R. Motwani, “Approximate nearest neighbors: towards removing the curse of dimensionality,” in *ACM Symposium on Theory of computing*, 1998, pp. 604–613.

[18] D. Achlioptas, “Database-friendly Random Projections: Johnson-lindenstrauss With Binary Coins,” *Journal of Computer and System Sciences*, vol. 66, pp. 671–687, 2003.

[19] J. N. Laska, P. T. Boufounos, M. A. Davenport, and R. G. Baraniuk, “Democracy in action: Quantization, saturation, and compressive sensing,” *Applied and Computational Harmonic Analysis*, vol. 31, no. 3, pp. 429–443, Nov. 2011.

[20] Y. Plan and R. Vershynin, “One-bit compressed sensing by linear programming,” *Arxiv preprint arXiv:1109.4299*, 2011.

[21] ——, “Dimension reduction by random hyperplane tessellations,” *Arxiv preprint arXiv:1111.4452*, 2011.

[22] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, “Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors,” *Arxiv preprint arXiv:1104.3160*, Apr. 2011.

[23] H. Shao, T. Svoboda, and L. V. Gool, “ZuBuD : Zurich Buildings database for image based recognition,” Computer Vision Lab, Swiss Federal Institute of Technology, Switzerland, Tech. Rep. 260, Apr. 2003. [Online]. Available: http://www.vision.ee.ethz.ch/showroom/zubud/

[24] S. Lloyd, “Least squares quantization in PCM,” *Information Theory, IEEE Transactions on*, vol. 28, no. 2, pp. 129 – 137, Mar. 1982.