Quantized Embeddings: An Efficient and Universal Nearest Neighbor Method for Cloud-based Image Retrieval

Shantanu Rane, Petros Boufounos and Anthony Vetro

Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA.

ABSTRACT

Efficient cloud-based image retrieval requires image features that have low communication overhead, provide an accurate representation of the image, remain robust to misalignment and scale, allow fast cloud-based matching, and require minimal updates to the parameters of the client's algorithm. Most image features spaces are only designed for accurate matching of particular classes of images. Thus, for example, a feature space that is discriminative for searching through images of buildings may not be as useful for discriminating medical images. Furthermore, the most discriminative feature spaces typically incur a large communication overhead.

In this paper, we propose a rate-efficient, feature-agnostic approach to encode the features. We extract quantized random projections of the image features under consideration, transmit these to the cloud server, and perform matching in the space of the quantized projections. The advantage of this approach is that, once the underlying feature extraction algorithm is chosen for maximum discriminability and retrieval performance (e.g., SIFT, or eigen-features), the random projections guarantee a rate-efficient representation and fast server-based matching with negligible loss in accuracy. Using the Johnson-Lindenstrauss Lemma, we show that pair-wise distances between the underlying feature vectors are preserved in the corresponding quantized embeddings. We report experimental results of image retrieval on two image databases with different feature spaces; one using SIFT features and one using face features extracted using a variant of the Viola-Jones face recognition algorithm. For both feature spaces, quantized embeddings enable accurate image retrieval combined with improved bit-rate efficiency and speed of matching, when compared with the underlying feature spaces.

Keywords: Randomized embeddings, nearest neighbors

1. INTRODUCTION

The amount of visual data generated by human beings continues to grow at a very rapid pace. This has created a plethora of new applications that were uncommon even a decade ago: Face tagging in images uploaded to social networking sites, extraction of rich information about products photographed at a supermarket, geographical and historical data mining about landmarks photographed on a touristic excursion, enhancing driving experience by means of windshield overlays, to name a few. The increased diversity and redundancy of today's rapidly growing databases enables the development of robust and novel applications with unprecedented capabilities. Unfortunately, the sheer size of the data makes image-based querying extremely challenging, especially in bandwidth-restricted applications that depend upon the speed of information retrieval.

Motivated by this problem, in this paper, we examine a fundamental and frequently used primitive in visual inference, namely, nearest neighbor (NN) matching. In particular, we are interested in fast and accurate approaches to NN matching, which are bandwidth-efficient, scalable and parallelizable. The method we propose is universal, in the sense that its design makes no assumption about the signals in the database or the query. Thus, while we focus on visual inference and image retrieval, it is immediately applicable to a wide variety of applications.

1.1 Desiderata for efficient and accurate NN-based Visual Inference

It goes without saying that the visual inference mechanism must identify nearest neighbors accurately. However, the accuracy requirement cannot be considered in isolation, especially when tradeoffs need to be made to ensure practical feasibility of the algorithm. The following requirements must also be satisfied:

Further author information: Send correspondence to Anthony Vetro, E-mail: avetro@merl.com, Telephone: 1-617-621-7591

1. *Compact upload from the client device to the server or cloud*: To minimize the communication overhead the visual inference mechanism should ensure that the client sends the query signal to the server using the smallest possible number of bits. Since the application deals with visual data, i.e., images or video, it is tempting to use a standardized compression algorithm to reduce the size of the query signal. Modern compression algorithms such as H.264/AVC, or JPEG2000 attempt to provide good rate distortion tradeoffs; their goal is to provide excellent reconstructed signal quality under a tight bit budget. However, this is not the best approach to NN search because the objective is not to reconstruct or display the signal, but rather to identify its nearest neighbors. Our work shows that this goal can be accomplished with a significantly smaller number of bits compared with that required by standardized compression algorithms.

2. *Robustness to variations in the visual query*: For most interesting applications of visual inference, there is no guarantee that the images taken by the client device are optimally aligned with the images in the server's database. For example, the server's database may be compiled via crowdsourcing from a very large number of users. Therefore, it is imperative that the NN search be robust to variations in the input image, such as, changes in translation, rotation, scaling, image quality, illumination, and so on.

3. *Fast NN search algorithm at the server*: In several application scenarios, the information retrieved by the visual inference mechanism may be time-sensitive. Examples of time-sensitive use cases include augmented reality-enabled headsets for people browsing museum exhibits, cars navigating traffic by means of turn-by-turn directions, etc. In such cases, it is beneficial for the server-based matching algorithm to be as fast and parallelizable as possible without compromising the matching performance.

4. *Futureproof algorithm*: In many visual inference applications, the server's database keeps changing as new visual data is added and low-quality or irrelevant data is discarded. For instance, the performance of a mobile phone-based augmented reality application would improve as a richer variety of images are accumulated at the database server with time. From a practical perspective, it is desirable that the algorithm and parameters used by the client remain unchanged when the server's database changes. Frequently changing the parameter values may well guarantee optimal performance, but would also require the client device to download frequent software updates containing the retrained parameters.

1.2 Related Work

There is a large body of work on image descriptors that enable fast and accurate image retrieval. Foremost among these are SIFT,[1] SURF,[2] GIST,[3] BRISK[4] and FREAK.[5] Of these, GIST captures global properties of the image, while the others capture local details at several salient points in an image, and therefore, have been used to match local features or patches. These descriptors can be used for image matching, registration and retrieval by combining hypotheses from several image patches, for example, using the popular Bag-of-Features approach.[6] 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.[7] However, recent work has reported FREAK to outperform SIFT in terms of robustness and speed.[5]

Nominally, a SIFT feature vector for a single salient point in an image is a real-valued, unit-norm 128-dimensional vector. Therefore, a prohibitively large bit rate is 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. Several methods have been proposed to compress the image descriptors and facilitate fast matching.[8–12] These methods—based on machine learning algorithms—use some form of classical or modern training-based techniques such as spectral hashing, Principle Component Analysis (PCA) or Linear Discriminant Analysis (LDA) to generate compact descriptors from the image descriptors such as SIFT or GIST. As mentioned above, while training-based methods can achieve accurate image retrieval, they are unsuited in applications where the database and the image can keep changing, necessitating repeated expensive training as new landmarks, products, etc. are added. 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.[13] In this method, gradient distributions are explicitly compressed, resulting in low-rate scale invariant descriptors.

Two other techniques[14,15] have been proposed for efficient remote image matching based on a version of Locality Sensitive Hashing (LSH,[16]), which is computationally simpler, but less bandwidth-efficient than CHoG, and does not need training. These techniques compute random projections of scale invariant features followed by one-bit quantization based on the sign of the random projections. By construction, as the quantizer makes a 1-bit decision, these works do not consider the tradeoff between dimensionality reduction and quantization. In a recent paper,[17] this tradeoff has been investigated and it has been shown both theoretically and experimentally that finer quantization with fewer random projections can be more bandwidth-efficient.

2. NN SEARCH BASED ON QUANTIZED EMBEDDINGS OF FEATURE SPACES

We are interested in an efficient, general method for NN search on visual data, such as images and videos. For the NN search to be accurate, it is necessary to choose an appropriate feature space with a high matching accuracy. However, the requirement of choosing a feature space for the highest matching accuracy presents a challenge in terms of efficient design; specifically, the features that provide the best NN matching performance are not in general the features that provide the best compression performance. This challenge informs our general approach to developing a new NN search method.

2.1 General Approach for NN Search on Visual Data

The first step involves deciding a feature space which enables accurate NN search. This step is usually easy because a rich literature exists on purpose-built visual descriptors. For example, for NN search of natural images, localized descriptors such as SIFT,[1] SURF,[2] Harris corner detectors,[18] etc., have been known to be very accurate. On the other hand, for NN search in the context of face recognition, other features such as eigenfaces[19] or Viola-Jones face descriptors[20] may be preferable. In still other applications such as NN search for fingerprint biometrics, the feature spaces may be specifically designed to maximize matching performance for a given biometric sensor.[21] In fact, for many applications, the state of the art has sufficiently matured that a specification of the particular matching problem immediately suggests the most suitable feature space. Thus, this is not the main focus of this paper.

Having chosen the feature space, the task is to transform these features into a descriptor that occupies a significantly smaller number of bits, while preserving the matching accuracy of the native feature space. For example, if a large number of SIFT features are extracted from an image, then the total bit rate occupied by the SIFT features may exceed the size of the JPEG-compressed version of the image. Even with a compression algorithm operating on the SIFT descriptors, it is often not possible to make the bit rate low enough to allow a compact and fast upload.

We propose to reduce the bit rate of the visual descriptors in two stages: First, perform a randomized embedding of the original feature vectors, which usually results in a reduction in the dimensionality as elaborated below. Second, perform scalar quantization of each element of the randomized embedding resulting in a small vector of quantization indices that can be efficiently transmitted to the server. In this work, we cover only the application of uniform scalar quantization of the embeddings. However, more exotic quantizer designs are possible within this framework, and have been shown to provide interesting tradeoffs among matching accuracy, bit rate efficiency, and privacy.[22,23]

2.2 Theoretical Foundations of Quantized Embeddings

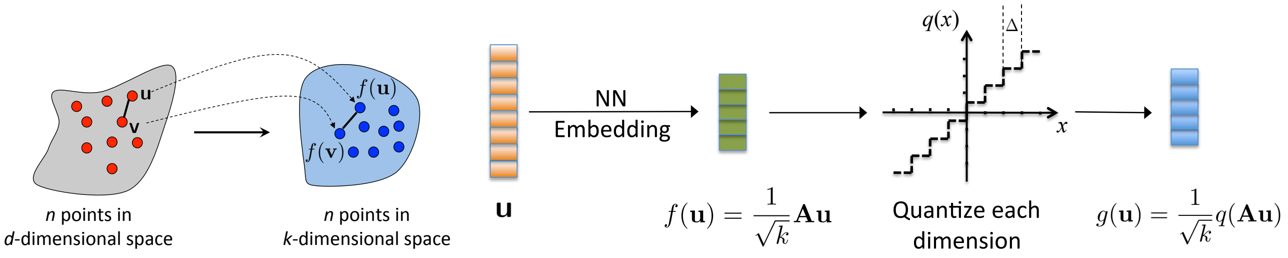
Our work relies on a randomized low-dimensional embedding of the features, as extracted from the image or video of interest. The use of embeddings is justified by the Johnson-Lindenstrauss (J-L) lemma, which forms the starting point of our theoretical development.

Theorem 1 *(Johnson-Lindenstrauss Lemma[24]) 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 $\mathbf{u}, \mathbf{v} \in \mathcal{X}$,

$$(1-\epsilon)\|\mathbf{u}-\mathbf{v}\|_2^2 \leq \|f(\mathbf{u}) - f(\mathbf{v})\|_2^2 \leq (1+\epsilon)\|\mathbf{u}-\mathbf{v}\|_2^2$$



(a) Nearest neighbor (NN) Embeddings

(b) Quantized NN Embeddings

Figure 1. (a) The Johnson-Lindenstrauss Lemma guarantees the existence of a NN embedding that preserves pairwise Euclidean distances. (b) A quantized NN embedding is derived by first obtaining a Johnson-Lindenstrauss embedding by multiplying the vectors in the canonical feature space by a random matrix, followed by scalar quantization of each element in the vector of randomized measurements.

In the above result and in the following development, $\|\cdot\|_p,\ p = 1, 2$ represents the ℓ_p 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 proportional to $\ln n$, the J-L 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, as shown in Fig 1(a).

The J-L lemma is extremely useful for querying huge databases (i.e., large n) with several attributes (i.e., large d). It suggests that seemingly high-dimensional visual data such as hi-resolution image and video signals reside on a manifold of much lower dimensionality than (say) the number of pixels.

One way to construct the embedding function f is to project the points from $\mathcal{X}$ onto a random hyperplane passing through the origin, drawn from a rotationally invariant distribution. In practice, this is accomplished by multiplying the data vector with a matrix whose entries are drawn from a specified distribution. For example, a matrix with i.i.d. $\mathcal{N}(0, 1)$ entries provides the distance-preserving properties in Theorem 1 with high probability. The following result[25,26] 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 i.i.d. from a $\mathcal{N}(0, 1)$ distribution. Let there be a set $\mathcal{X} \subset \mathbb{R}^d$ that contains n points. Then, the mapping $f(\mathbf{u}) = \frac{1}{\sqrt{k}}\mathbf{Au}$ satisfies the distance preserving property in Theorem 1 for all pairs $\mathbf{u}, \mathbf{v} \in \mathcal{X}$, 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 has also been shown to hold for other distributions on $a(i, j)$ besides the normal distribution. For example, the theorem holds when the $a(i, j)$ are drawn i.i.d. from a Rademacher distribution, i.e., taking values ±1 with equal probability.[27] In what follows, however, we consider only the normal gaussian case.

Following the terminology used in compressed sensing literature, we refer to each element of the embedding $f(\mathbf{u})$ as a randomized measurement of the signal $\mathbf{u}$. When it is convenient, we also sometimes refer to each element of the embedding $f(\mathbf{u})$ as a random projection of the signal $\mathbf{u}$; the words "projection" and "measurement" are henceforth interchangeable. Furthermore, in our subsequent development the vectors $\mathbf{u}, \mathbf{v} \in \mathcal{X}$ represent feature vectors obtained by applying the chosen feature extraction algorithm to the query image or video signal. Depending upon the application and the chosen feature extraction algorithm, a given input image or video may be processed to yield one or more such feature vectors.

Even though the dimensionality of $f(\mathbf{u})$ is smaller than that of $\mathbf{u}$, the elements of $f(\mathbf{u})$ are real-valued and thus cannot be represented using a finite number of bits. In order to make it feasible to store and transmit the distance-preserving embeddings $f(\mathbf{u})$, the real-valued random projections have to be quantized. We are particularly interested in the distance-preserving property of *quantized* embeddings, specifically 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 $1 - n^{-\beta}$. We examine the question: 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[17] 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*

$$(1 - \epsilon)\|\mathbf{u} - \mathbf{v}\|_2 - \Delta \leq \|g(\mathbf{u}) - g(\mathbf{v})\|_2 \leq (1 + \epsilon)\|\mathbf{u} - \mathbf{v}\|_2 + \Delta \tag{2}$$

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

The above result indicates that quantized embeddings preserve pairwise Euclidean distance up to a multiplicative factor $1 \pm \epsilon$, and an additive factor $\pm\Delta$. For the best embedding accuracy, we would like to simultaneously reduce both ϵ and Δ. Note however that, by Theorem 1, reducing ϵ necessitates increasing the number of randomized measurements k. Furthermore, reducing Δ amounts to increasing the number of bits allocated to each measurement, and therefore the total number of bits allocated to the embedding $f(\cdot)$. This suggests that, if the total bit budget allocated for representing the embedding $f(\cdot)$ is fixed, we cannot simultaneously reduce the embedding error factor, ϵ, and quantization error, Δ. We now explore this tradeoff and its consequences in further detail.

The quantization interval Δ depends on the design of the scalar quantizer and the bit-rate B used to encode each coefficient. We consider a finite uniform scalar quantizer, as shown in Fig. 1(b), 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, k, and number of bits per measurement, R/k, becomes more explicit:

$$(1 - \epsilon)\|\mathbf{u} - \mathbf{v}\|_2 - 2^{-\frac{R}{k}+1}S \leq \|g(\mathbf{u}) - g(\mathbf{v})\|_2 \leq (1 + \epsilon)\|\mathbf{u} - \mathbf{v}\|_2 + 2^{-\frac{R}{k}+1}S, \tag{3}$$

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 above theoretical development: non-uniform quantization and saturation. A non-uniform scalar quantizer, tuned to the distribution of the measurements, may improve embedding performance. A nearly optimal non-uniform quantizer can be designed, for example using the Lloyd-Max algorithm. It will still suffer the same tradeoff between number of bits per measurement and number of measurements, though the exact tradeoff at low rates is difficult to determine and beyond our current scope. At high rates, it is known that uniform quantization is optimal, i.e., when the Lloyd-Max algorithm is run using the randomized measurements as input data, the resulting quantization cells will look nearly uniform. Thus, the theory discussed above still applies at high rates.

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.[28] 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. In this work we do not tune the saturation level; instead we measure the maximum and minimum values taken by the randomized measurements and conservatively choose the positive and negative ranges ($\pm S$) of the quantizers such that saturation occurs with negligibly small probability.

It should also be noted that while the above theoretical development holds for uniform quantization with more than 2 quantization levels, it partially breaks down for quantization at 1-bit per measurement, i.e., when only the sign of the randomized measurement is retained. In particular, if one signal in the set of interest is a positive scalar multiple of another, then these two signals will be indistinguishable under the randomized embedding. While the distance preservation 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 preservation of the angles between two signals—i.e., their correlation or inner product—instead of the distance between them.[29–31]

2.3 Extension to NN Search based on Non-Euclidean Distance Measures

Thus far, we have considered NN search for quantized, randomized embeddings of visual data under a Euclidean (ℓ_2) distance criterion. However, there are several feature extraction algorithms in which signal comparison is based on comparing the feature vectors under different distance criteria. For example, gene sequences are best compared using edit (Levenshtein) distance.[32] Face features extracted using Viola-Jones style face detectors[20] provide better matching under the ℓ_1 distance criterion.

Computing distance preserving embeddings for any distance criterion is a challenging problem. However, embeddings for specific distance measures, such as ℓ_1 distance[33] and edit distance,[34–36] do exist. These embeddings provide distance preservation properties similar to the ones that Johnson-Lindenstrauss Lemma provides for ℓ_2, though the guarantees are substantially weaker. For example, Indyk *et al.*[33] have described an embedding into a normed ℓ_1 metric space that preserves ℓ_1 distance, such that the distance in the embedding space within a $1-\epsilon$ factor of the original distance with *high* probability, and within a $1+\epsilon$ factor of the original distance with *constant* probability. It has been shown that a stronger guarantee, such as preserving distances to within a $1 \pm \epsilon$ factor with high probability, as given by Johnson-Lindenstrauss, does not exist for ℓ_1 distance.[37] To the best of our knowledge, the NN matching performance for quantized versions of these embeddings has not yet been reported.

In this paper, we employ a different method to achieve ℓ_1 distance preserving embeddings. Our method is applicable to scenarios in which feature extraction algorithms operating on visual data result in integer feature vectors, where the maximum and minimum values of the feature elements are known. For such cases, the integer vectors can be naively mapped into binary feature vectors which are elements of a real-valued ℓ_2 metric space, such that the squared ℓ_2 distance between the binary vectors is equal to the ℓ_1 distance between the original integer feature vectors.[17,38]

In particular, without loss of generality, consider that each element $u_i, i \in \{1, 2, \ldots, d\}$ of the integer feature vector $\mathbf{u}$ takes values between 0 and V. If the minimum and/or maximum values are negative, it is always possible to shift the values, such that they take values in the set $\{0, 1, ..., V\}$. Next, for all $i \in \{1, 2, \ldots, d\}$ obtain $b(u_i) \in \{0, 1\}^V$, where $b(x)$ is simply a binary sequence containing 1's in its first x locations and 0's in the following $V - x$ locations. Concatenate all the $b(u_i)$ to obtain a binary vector which—by a slight abuse of notation—we denote as $b(\mathbf{u})$. By construction $b(\mathbf{u}) \in \{0, 1\}^{Vd}$.

As a consequence of this simple mapping, the following holds for all $\mathbf{u}, \mathbf{v} \in \{0, 1, \ldots, V\}^d$:

$$\|\mathbf{u} - \mathbf{v}\|_1 = \|b(\mathbf{u}) - b(\mathbf{v})\|_1 = \|b(\mathbf{u}) - b(\mathbf{v})\|_2$$

The result is a mapping that preserves the pairwise ℓ_1 distance between the original integer feature vectors. Note that the mapping considerably increases the dimensionality of the problem from d to Vd. However, note that the total number of points in the set remains the same; in other words, even in the space of the binary mappings, the maximum possible number of points is still $(V + 1)^d$ and not 2^{Vd}. Note that $V + 1 \ll 2^V$. Therefore, by applying a dimensionality reducing embedding, such as the quantized embedding in Proposition 1, we can map the vectors $b(\mathbf{u}), b(\mathbf{v}) \subset \{0, 1\}^{Vd}$ into lower-dimensional vectors $g(b(\mathbf{u}))$, $g(b(\mathbf{v})) \in \mathbb{R}^k$, respectively, where $k < d$. Furthermore, by choosing the quantization step size Δ and the number of randomized measurements k appropriately, it may be possible ensure that total bit rate required to store and transmit the embeddings $g(b(\mathbf{u}))$ is significantly lower than that $d \log_2 V$, which is the bit rate required to transmit the underlying feature vectors. Our experiments on a real-world database of face images, presented in Section 4, demonstrate that this is indeed the case.

3. NEAREST NEIGHBOR SEARCH USING QUANTIZED EMBEDDINGS

To retrieve information about a query object, a user captures an image or a video of the query object, extracts the appropriate features and transmitting randomized measurements of those features to a database server. The server executes the query on the database to find the closest matching image(s) or video(s) of the query object according to a predetermined distance criterion and transmits rich information about that object back to the user. This rich information may consist of image-specific metadata, geographical or historical information, identification information about a person or face in the image, graphical overlays for augmented reality, and so on. The algorithms executed by the server and the client for determining the NN of the query object are described below.

3.1 Database preparation at the server

The server generates the matrix $\mathbf{A}$ and specifies the quantization step size to be used. To build the database, it acquires a set of images $\mathbf{I}_1, \ldots, \mathbf{I}_T$ of S objects, where $S \leq T$. For each object, the server obtains or generates application-specific metadata, $\mathbf{D}_s, s \in \{1, ..., S\}$. Then, given a feature space suitable for the application under consideration, it applies a feature extraction algorithm on each image $\mathbf{I}_t$ to generate one or more feature vectors from each image. The number of features obtained from each image depends on the application as well as variables such as the scene content, the illumination and the resolution of the sensor capturing the picture. Let L denote the number of feature vectors extracted from all images of all objects and $\mathbf{y}_l, l = 1, \ldots, L$ denote each feature vector; typically, $L \gg S$. Using these L feature vectors, the server computes the database $\{g(\mathbf{y}_1), \ldots, g(\mathbf{y}_L)\}$, where each $g(\mathbf{y}_i)$ is an R-bit quantized embedding of $\mathbf{y}_i$. As a final book-keeping step, the server generates a lookup table $\lambda(l) \subset \{1, ..., S\}, l = 1, \ldots, L$ where each $\lambda(l)$ indexes the object from which the vector $g(\mathbf{y}_l)$ (or equivalently $\mathbf{y}_l$) was extracted.

3.2 Client query

The client obtains the embedding matrix $\mathbf{A}$ and the quantization step size Δ from the server. To enable this in practice, the distribution of the $a(i, j)$ is approximated by a pseudorandom number generator. The seed of this pseudorandom number generator is sent to the client 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 realizations of $\mathbf{A}$. Note that, though matrix $\mathbf{A}$ is initially chosen at random in this way by the server, it is fixed for the remainder of the procedure. As a consequence of the Johnson-Lindenstrauss Lemma, a randomly chosen matrix will provide good distance preservation with a very high probability. Once the client acquires the query image it executes the relevant feature extraction algorithm to derive a set of features $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$. Using these $M \geq 1$ features and the matrix $\mathbf{A}$, the client computes and transmits to the server the corresponding quantized embeddings $\{g(\mathbf{x}_1), \ldots, g(\mathbf{x}_M)\}$.

3.3 Approximate NN search at the server and meta-data retrieval

The server receives $\{g(\mathbf{x}_1), \ldots, g(\mathbf{x}_M)\}$ from the client. For each of the $g(\mathbf{x}_m)$ it computes the nearest neighbor among its database, i.e., among $\{g(\mathbf{y}_1), \ldots, g(\mathbf{y}_L)\}$. The result is M nearest neighbor pairs, one pair for each embedding $g(\mathbf{x}_m)$. Out of these M pairs, the server chooses the $J \leq M$ pairs $\{g(\mathbf{x}_{(j)}), g(\mathbf{y}_{(j)})\}, j = 1, 2, ..., J$ that are closest in embedding distance. For each of the J pairs, the server uses the lookup table $\mathbf{\Lambda}$ to read off the index of the object from which feature vector $\mathbf{y}_{(j)}$ was derived, storing it in $\alpha_j \in \{1, ..., s\}$. The object s_0 most common among α_j, i.e., the one with the largest number of nearest neighbor matches among the J best matching features, is the response to the query; its metadata are returned to the client.

4. EXPERIMENTAL RESULTS

In this section, we discuss the performance of NN search in two application scenarios. The first application is visual inference on natural images: we find similar images based on embeddings of SIFT features extracted from the client's image. Thus, efficient NN matching is carried out while preserving the advantages of SIFT vectors, namely robustness to translation, rotation and scale, and without compromising the matching accuracy.

The second application identifies faces from a variety of face images, which contain differences in pose, illumination, hair-style, etc. In this case, the underlying feature space consists of features similar to Viola-Jones face features, best matched using the ℓ_1 distance metric.[20] As described in Sec. 2, the features are binarized to map the ℓ_1 distance into Euclidean distance and, subsequently embedded into a lower-dimensional subspace. Our experiments indicate that quantized embeddings approach the matching accuracy of underlying Viola-Jones features at a fraction of the bit rate.

4.1 Embeddings of Scale-Invariant Features

We conducted experiments on a public database to evaluate the performance of meta-data retrieval using quantized embeddings of scale-invariant features. We used the ZuBuD database,[39] 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.

As mentioned above, we extract the popular SIFT features from the client's and server's images and match the features using their quantized embeddings. The exact details, including the matching algorithm and extensive results, have been described in a recent paper.[17] Here, we summarize the experiments that examine the performance of our approach and the trade-off between number of measurements and number of bits per measurement with respect to that performance.

To measure the fidelity of the algorithm, we define the probability of correct retrieval P_{cor} simply as the expected value of the ratio of the number of query images for which the NN search yields the correct match (N_c), to the total number of query images (N_q), which is 201 for the ZuBuD database. In this definition, 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, reporting the mean of the ratio N_c/N_q as P_{cor}.

We compared the accuracy of meta-data retrieval achieved by the LSH-based schemes[14,15] with our multi-bit quantization approach. Both the LSH-based schemes use random projections of the SIFT vectors followed by 1-bit quantization according to the sign of the random projections. Fig. 2(a) shows the variation of P_{cor} against the number of projections for the LSH-based schemes. This is significantly outperformed by 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. For the same number of measurements, this comes at a significant rate increase, compared to 1-bit measurements.

Next, we examine experimentally the optimal trade-off between number of measurements, k, and bits per measurement, B, to achieve highest rate of correct retrieval, P_{cor}, given a fixed total rate budget, $R = kB$, per embedded descriptor. This is shown in Fig. 2(b).A 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. However, more bits per measurement are not always better. In particular, the 3 and 4-bit quantizers provide the highest P_{cor} for a given bit budget, outperforming the 5-bit quantizer.

Using quantized embeddings is significantly more efficient than sending quantized versions of the original descriptor. In our experiments, the performance of quantized SIFT vectors saturated at 94%, using 384 bits per descriptor. The same performance is achieved with quantized embeddings of SIFT descriptors using only 80 bits per descriptor. Furthermore, the scheme is much more efficient than compressing the original image via JPEG and transmitting it to the server for SIFT-based matching. At 80 quality factor, the average size of a JPEG-compressed image from the ZuBuD database is 58.5 KB. In comparison, the average total bit rate of all quantized embeddings computed for an image in this database is 2.5 KB.

4.2 Embeddings of Face Features

In our next experiment, quantized randomized embeddings were used for NN search of face images. Again, the underlying assumption that pixels of face images or features extracted from face images lie on a low-dimensional manifold and little or no performance degradation would result if the faces were matched in this lower dimensional subspace.

A total of 2752 face images, corresponding to 292 persons were obtained from the Multiple Biometric Grand Challenge (MBGC) database for this experiment. There are 5 or more images available for each person, of which one was randomly chosen to be the query image, while the other 4 were considered as images stored in the server's database. The images vary widely in pose, makeup or hairstyle of the subject, illumination, etc. making face detection and recognition a challenging task. Analogous to SIFT vectors in the previous section, we employed the face feature extraction techniques based on a variant of the popular Viola-Jones algorithm.[20]

The experiment, however, differs from that in the previous section in three respects: Firstly, rather than extracting several SIFT descriptors per image, only one feature vector is extracted per face image. This vector consists of 900 integers, each taking a value between 0 and 255. Secondly, rather than performing NN search under a Euclidean distance

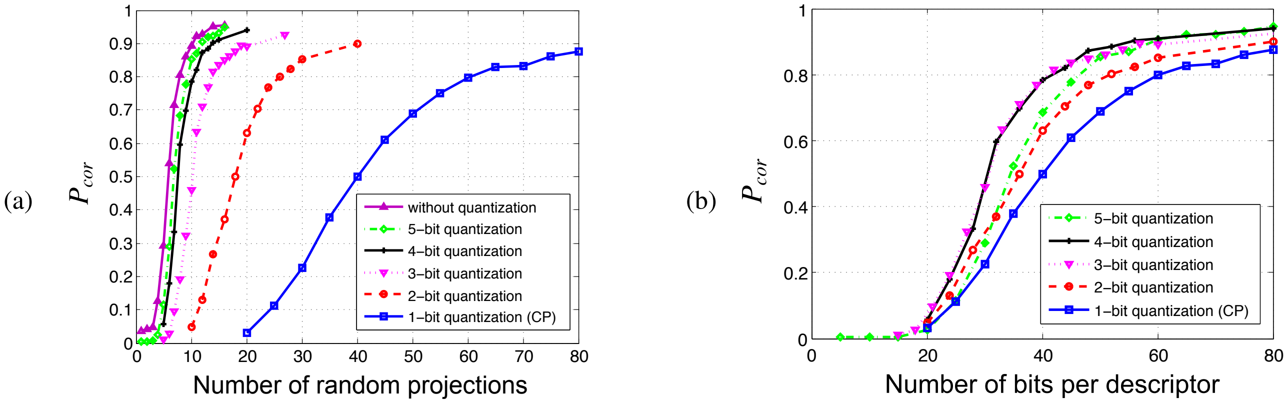


Figure 2. (a) Multi-bit quantization with fewer random projections outperforms LSH-based schemes[14, 15] which employ 1-bit quantization with a large number of random projections. (b) When the bit budget allocated to each descriptor (vector) is fixed, the best retrieval performance is achieved with 3-bit and 4-bit quantization.

criterion, these features vectors give the most accurate face recognition performance when the NN search is carried out under a Manhattan (ℓ_1) distance criterion. The integer feature vectors extracted from the face images are binarized, and quantized embeddings of these binarized features used for the NN search. Thirdly, the NN search is performed slightly differently from the previous experiment. Specifically, since there is only one feature vector per face image, the server finds the J closest images using the quantized embeddings. Then it finds the $K \leq J$ identities corresponding to these J closest images, and chooses the most frequently occurring identity, i.e., the mode of the identities, as the NN match. If $K = J$, one of the matching identities is chosen at random as the NN match. In our experiments, we set $J = 10$.

The face recognition accuracy is measured as above using the empirical probability of correct retrieval P_{cor}. Once again, P_{cor} is computed as the average of 25 readings of relative frequency of correct face recognition, where the average is computed over the randomness introduced in the computation of the embeddings, i.e., over 25 realizations of the matrix $\mathbf{A}$. We compared the accuracy of meta-data retrieval achieved by the LSH-based schemes, that use random projections of the Viola-Jones feature vectors followed by 1-bit quantization according to the sign of the random projections. Fig. 3(a) shows the variation of P_{cor} against the number of projections for the LSH-based schemes as well as the multibit quantization schemes. As was the case for the ZuBuD database, the multibit quantization schemes provide a significantly improved face recognition accuracy than the LSH-based schemes. Using 6 bits per dimension nearly achieves the performance of unquantized random projections.

Next, as with natural images and SIFT descriptors, we examine the optimal trade-off between number of measurements and bits per measurement that achieves the highest probability of correct retrieval for each embedded Viola-Jones descriptor. The results are shown in Fig. 3(b). The multibit quantizers again provide higher face recognition accuracy than the LSH-based schemes, confirming that fewer finely quantized projections outperform a many coarsely quantized projections. Particularly, the 3-bit and 4-bit quantizers provide the best trade-off, i.e., the highest P_{cor} for a given bit budget, outperforming the 5-bit and 6-bit quantizers. Interestingly, the best trade-off is at the same number of bits per projection, $B = 3$ or 4, as the previous experiment. Whether this is trade-off universal is an open theoretical question.

5. SUMMARY

In this paper we described an embeddings-based method to reduce the rate required to query a database. Our approach uses well-established feature extraction methods and encapsulates them within quantized Johnson-Lindenstrauss embeddings. Thus, we exploit the finely tuned application-specific retrieval performance that established features can provide, as well as the bit-rate reduction afforded by quantized embeddings. These benefits can be confirmed experimentally for two different databases, one containing urban images, and another containing human faces.

Experiments with both datasets confirm the theoretical tradeoff observed between the fidelity of representing each randomized measurement in the embedding and the number of randomized measurements. Different from our past work, the experiments in this paper confirm that this tradeoff holds even when the matching criterion is the pairwise ℓ_1 distances between the underlying image features, provided the quantized embeddings are still matched based on the ℓ_2 distance.

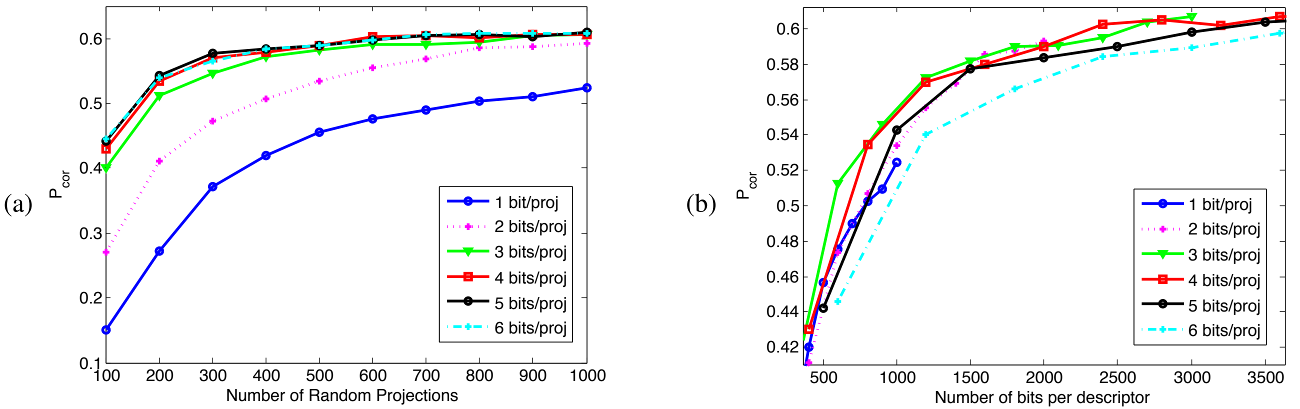


Figure 3. (a) Multi-bit quantization with fewer random projections outperforms 1-bit quantization[14, 15] with a large number of random projections. (b) For a fixed bit budget allocated per descriptor, the best retrieval performance is achieved with 3-bit and 4-bit quantization. Interestingly, this result for face recognition based on Viola-Jones features is similar to that obtained with urban image matching based on SIFT descriptors.

ACKNOWLEDGMENTS

The authors would like to acknowledge the efforts of Mu Li from Pennsylvania State University for his contributions to our early research efforts in quantized embeddings.[17]

REFERENCES

1. D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision* **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* **110**(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* **42**, pp. 145–175, 2001.
4. S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2548–2555, IEEE, 2011.
5. A. Alahi, R. Ortiz, and P. Vandergheynst, "Freak: Fast retina keypoint," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 510–517, IEEE, 2012.
6. J. Sivic and A. Zisserman, "Efficient visual search of videos cast as text retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence* **31**, pp. 591–606, Apr. 2009.
7. K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence* **27**, pp. 1615 –1630, Oct. 2005.
8. 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*, pp. 1 –8, June 2008.
9. 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., pp. 1753–1760, 2009.
10. 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* **PP**(99), p. 1, 2011.
11. C. Yeo, P. Ahammad, and K. Ramchandran, "Coding of image feature descriptors for distributed rate-efficient visual correspondences," *International Journal of Computer Vision* **94**, pp. 267–281, 2011. 10.1007/s11263-011-0427-1.
12. C. Strecha, A. Bronstein, M. Bronstein, and P. Fua, "LDAHash: Improved matching with smaller descriptors," *IEEE Trans. Pattern Analysis and Machine Intelligence* **34**, pp. 66 –78, Jan. 2012.
13. 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* **96**, pp. 384–399, 2012.
14. C. Yeo, P. Ahammad, and K. Ramchandran, "Rate-efficient visual correspondences using random projections," in *IEEE International Conference on Image Processing*, pp. 217 –220, Oct. 2008.

15. 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*, pp. 3477 –3484, June 2010.
16. A. Andoni and P. Indyk, "Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions," *Commun. ACM* **51**, pp. 117–122, Jan. 2008.
17. M. Li, S. Rane, and P. Boufounos, "Quantized embeddings of scale-invariant image features for mobile augmented reality," in *IEEE International Workshop on Multimedia Signal Processing*, (Banff, Canada), September 17-19 2012.
18. C. Harris and M. Stephens, "A combined corner and edge detector.," in *Alvey vision conference*, **15**, p. 50, Manchester, UK, 1988.
19. M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience* **3**(1), pp. 71–86, 1991.
20. M. Jones and P. Viola, "Face recognition using boosted local features," *MERL Technical Report, TR2003-25* , May 2003.
21. A. K. Jain, P. J. Flynn, and A. A. Ross, *Handbook of biometrics*, Springer, 2008.
22. P. T. Boufounos and S. Rane, "Secure binary embeddings for privacy preserving nearest neighbors," in *Proc. Workshop on Information Forensics and Security (WIFS)*, (Foz do Iguau, Brazil), November 29 - December 2 2011.
23. S. Rane and P. T. Boufounos, "Privacy-preserving nearest neighbor methods: Comparing signals without revealing them," *IEEE Signal Processing Magazine* , March 2013.
24. W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary Mathematics* **26**, pp. 189 –206, 1984.
25. S. Dasgupta and A. Gupta, "An elementary proof of a theorem of Johnson and Lindenstrauss," *Random Structures & Algorithms* **22**(1), pp. 60–65, 2003.
26. P. Indyk and R. Motwani, "Approximate nearest neighbors: towards removing the curse of dimensionality," in *ACM Symposium on Theory of computing*, pp. 604–613, 1998.
27. D. Achlioptas, "Database-friendly Random Projections: Johnson-lindenstrauss With Binary Coins," *Journal of Computer and System Sciences* **66**, pp. 671–687, 2003.
28. 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* **31**, pp. 429–443, Nov. 2011.
29. Y. Plan and R. Vershynin, "One-bit compressed sensing by linear programming," *Arxiv preprint arXiv:1109.4299* , 2011.
30. Y. Plan and R. Vershynin, "Dimension reduction by random hyperplane tessellations," *Arxiv preprint arXiv:1111.4452* , 2011.
31. 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.
32. D. Gusfield, *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*, Cambridge University Press, 1997.
33. P. Indyk, "Stable distributions, pseudorandom generators, embeddings, and data stream computation," *Journal of ACM* **53**(3), pp. 307–323, 2006.
34. A. Andoni, M. Deza, A. Gupta, P. Indyk, and S. Raskhodnikova, "Lower bounds for embedding edit distance into normed spaces," in *Proceedings of the fourteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 523–526, 2003.
35. Z. Bar-Yossef, T. Jayram, R. Krauthgamer, and R. Kumar, "Approximating edit distance efficiently," in *Foundations of Computer Science, 2004. Proceedings. 45th Annual IEEE Symposium on*, pp. 550–559, IEEE, 2004.
36. R. Ostrovsky and Y. Rabani, "Low distortion embeddings for edit distance," *Journal of the ACM (JACM)* **54**(5), p. 23, 2007.
37. B. Brinkman and M. Charikar, "On the impossibility of dimension reduction in l 1," *Journal of the ACM (JACM)* **52**(5), pp. 766–788, 2005.
38. W. Lu, A. Varna, and M. Wu, "Secure Image Retrieval through Feature Detection," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, (Taipei, Taiwan), 2009.
39. H. Shao, T. Svoboda, and L. V. Gool, "ZuBuD : Zurich Buildings database for image based recognition," Tech. Rep. 260, Computer Vision Lab, Swiss Federal Institute of Technology, Switzerland, Apr. 2003.