MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Efficient Coding of Signal Distances Using Universal Quantized Embeddings

Boufounos, P.T.; Rane, S.

TR2013-009 March 2013

Abstract

Traditional rate-distortion theory is focused on how to best encode a signal using as few bits as possible and incurring as low a distortion as possible. However, very often, the goal of transmission is to extract specific information from the signal at the receiving end, and the distortion should be measured on that extracted information. In this paper we examine the problem of encoding signals such that sufficient information is preserved about their pairwise distances. For that goal, we consider randomized embeddings as an encoding mechanism and provide a framework to analyze their performance. We also propose the recently developed universal quantized embeddings as a solution to that problem and experimentally demonstrate that, in image retrieval experiments, universal embedding can achieve up to 25% rate reduction over the state of the art.

Data Compression Conference (DCC)

Efficient Coding of Signal Distances Using Universal Quantized Embeddings

Petros T. Boufounos and Shantanu Rane
Mitsubishi Electric Research Laboratories
Cambridge, MA 02139, USA.
{petrosb,rane}@merl.com

Abstract

Traditional rate-distortion theory is focused on how to best encode a signal using as few bits as possible and incurring as low a distortion as possible. However, very often, the goal of transmission is to extract specific information from the signal at the receiving end, and the distortion should be measured on that extracted information. In this paper we examine the problem of encoding signals such that sufficient information is preserved about their pairwise distances. For that goal, we consider randomized embeddings as an encoding mechanism and provide a framework to analyze their performance. We also propose the recently developed universal quantized embeddings as a solution to that problem and experimentally demonstrate that, in image retrieval experiments, universal embedding can achieve up to 25% rate reduction over the state of the art.

I. Introduction

Source coding theory and practice has primarily focused on how to best encode a signal for transmission using the fewest possible bits while incurring the smallest possible distortion. For example, in image or video compression, the encoder aims to reduce the bit-rate for a given visual reconstruction quality. This goal is dictated by the end user of the signal: an image or a video will be viewed by a human being. Quite often, however, the end user of a signal is not a human being observing the distorted signal per se, but a server extracting information. In this case, the goal is different: encoding must happen in a way that does not destroy the information that the server wants to extract, even if the signal itself cannot be completely recovered. In particular, we examine applications in which the server is interested in extracting only the information about the distance of a signal from its nearest neighbors.

This paper examines how to efficiently encode signals for transmission such that the receiver can approximately determine the distance between signals up to a specified radius. Our encoding exploits the recently developed theory for efficient universal quantization and universal quantized embeddings [1]. We demonstrate that, using universal quantized embeddings, we are able to improve compression performance up to 25% over previous embedding-based approaches [2], [3], including our own earlier work [4]. The main advantage of universal embeddings is that they preserve distance information only up to a certain radius, as required to determine the near neighbors, and not any farther. Thus, rate is not wasted in coding distances larger than necessary.

Our main—but not the only—motivating example is image retrieval, with emphasis on augmented reality (AR) applications. As we discuss in [4], AR and more general image retrieval applications can benefit significantly by efficient coding of distances to a signal's nearest neighbors. In typical cloud-based image retrieval applications, a client transmits

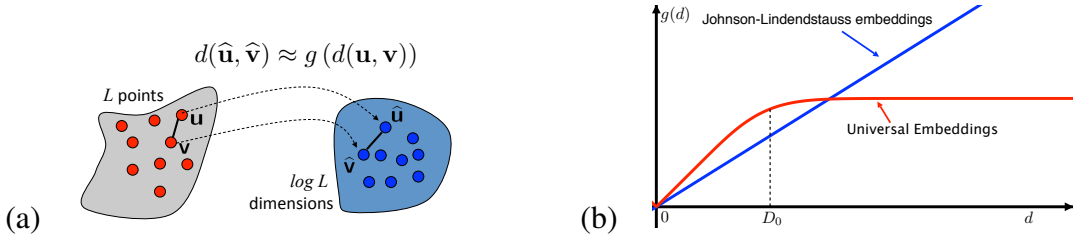

Fig. 1. (a) Distance-preserving embeddings approximately preserve a function $g(\cdot)$ of the distance, allowing distances to be computed in a space that (typically) has fewer dimensions and produce signals that often require lower transmission rate. (b) For most embeddings, such as JL Embeddings, this function is linear, as shown in blue. For the universal quantized embeddings discussed in this paper, the function is approximately linear initially and quickly flattens after a certain distance D_0, as shown in red.

to a cloud server a query photograph taken by the user, or features extracted from that photograph, requesting more information on the objects in the picture. The server extracts features if necessary, searches the database for tagged pictures with similar features and returns the requested information about the picture. This search should be quick and computationally efficient, while the transmission should be bandwidth-efficient.

Recent developments in image retrieval have significantly reduced its computational complexity. Image descriptors, such as SIFT [5], SURF [6], GIST [7] and related techniques, enable fast searches using global image characteristics or local image details if communication cost is not an issue. To further address communication complexity, several training-based methods have been developed [8]–[12], discussed in detail in [4]. However, these methods all require retraining whenever new database entries are added, causing a change in the signal statistics. In AR applications, re-training is undesirable; in addition to the complexity of training at the server, it repeatedly necessitates updating the client with the re-trained parameters. Thus, methods that do not require training are preferable. These include CHoG [13], in which the descriptors are explicitly designed to be compressed using vector quantization and compact projection [2], [3] which uses Locality Sensitive Hashing (LSH) [14] on established descriptors.

The next section contains a brief background on embeddings and universal scalar quantization. Section III discusses how embeddings can be used to efficiently encode signal distances and analyzes their performance. Section IV describes an augmented reality scheme that efficiently retrieves image-dependent meta data, by leveraging universal quantized embeddings. Section V experimentally demonstrates the rate reduction achieved using our approach to the AR problem, applied to a well-known public database [15]. Section VI discusses our findings and concludes the paper.

II. Background

A. Randomized Embeddings

An *embedding* is a transformation of a set of signals in a high-dimensional space to a (typically) lower-dimensional one such that some aspects of the geometry of the set are preserved, as depicted in Fig. 1(a). Since the set geometry is preserved, distance computations can be performed directly using the low-dimensional—and often low bitrate—embeddings, rather than the underlying signals.

The best known embeddings are the Johnson-Lindenstrauss embeddings [16]—functions $f : \mathcal{S} \to \mathbb{R}^K$ from a finite set of signals $\mathcal{S} \subset \mathbb{R}^N$ to a K-dimensional vector space such

that, given two signals $\mathbf{x}$ and $\mathbf{y}$ in $\mathcal{S}$, their images satisfy:

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\|_2^2 \leq \|f(\mathbf{x})-f(\mathbf{y})\|_2^2 \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\|_2^2.$$

In other words, these embeddings preserve ℓ_2 distances, i.e., Euclidean distances, of point clouds within a small factor, measured by ϵ.

Johnson and Lindenstrauss demonstrated that an embedding preserving the distances as described above exists in a space of dimension $K = O(\frac{1}{\epsilon^2}\log L)$, where L is the number of signals in $\mathcal{S}$ (its cardinality) and ϵ the desired tolerance in the embedding. Remarkably, K is independent of N, the dimensionality of the signal set $\mathcal{S}$. Subsequent work showed that it is straightforward to compute such embeddings using a linear mapping. In particular, the function $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$, where $\mathbf{A}$ is a $K \times N$ matrix whose entries are drawn randomly from specific distributions, is a J-L embedding with overwhelming probability. Commonly used distributions are i.i.d. Gaussian, i.i.d. Rademacher, or i.i.d. uniform.

A J-L embedding typically results in a significant dimensionality reduction. However, dimensionality reduction does not immediately produce rate reduction; the embeddings must be quantized for transmission and, if the quantization is not well designed, performance suffers [4]. In particular, J-L embeddings with scalar quantization satisfy

$$(1-\epsilon)\|\mathbf{x}-\mathbf{y}\| - \tau \leq \|f(\mathbf{x})-f(\mathbf{y})\| \leq (1+\epsilon)\|\mathbf{x}-\mathbf{y}\| + \tau,$$

where $\tau \propto 2^{-B}$ is the quantizer step size, decreasing exponentially with the number of bits used per dimension, B. On the other hand, ϵ is a function of K, the projection's dimensionality, and scales approximately as $1/\sqrt{K}$. In the extreme case of 1-bit scalar quantization the embedding does not preserve signal amplitudes and, therefore, their ℓ_2 distances. Still, it does preserve their angle, i.e., their correlation coefficient [17], [18].

When designing a quantized embedding, the total rate is determined by the dimensionality of the projection and the number of bits used per dimension: $R = KB$. At a fixed rate R, as the dimensionality K increases, the accuracy of the embedding before quantization, as reflected in ϵ, is increased. But to keep the rate fixed the number of bits per dimension should also decrease, which decreases the accuracy due to quantization, reflected in τ. This non-trivial trade-off is explored in detail in [4]; at a constant rate a multibit quantizer outperforms the 1-bit quantizers examined in earlier literature [2], [3].

B. Universal Quantization and Embeddings

Universal scalar quantization, first introduced in [1], fundamentally revisits scalar quantization and redesigns the quantizer to have non-contiguous quantization regions. This approach also relies on a Johnson-Lindenstrauss style projection, followed by scaling, dithering and scalar quantization:

$$f(\mathbf{x}) = Q(\Delta^{-1}(\mathbf{A}\mathbf{x} + \mathbf{w})), \tag{1}$$

where $\mathbf{A}$ is a random matrix with $\mathcal{N}(0,\sigma^2)$-distributed, i.i.d. elements, Δ^{-1}—abusing notation—an element-wise scaling factor, $\mathbf{w}$ a dither vector with i.i.d. elements, uniformly distributed in $[0,\Delta]$, and $Q(\cdot)$ a scalar quantizer operating element-wise on its input.

The breakthrough feature in this method is the modified 1-bit scalar quantizer, designed to have non-contiguous quantization intervals as shown in Fig. 2(a). The quantizer can be thought of as a regular uniform quantizer, computing a multi-bit representation of a signal and preserving only the least significant bit (LSB) of the representation. Thus,

scalar values in $[2l, 2l+1)$ quantize to 1 and scalar values in $[2l+1, 2(l+1))$, for any integer l, quantize to 0. Since $Q(\cdot)$ is a 1-bit quantizer, this method encodes using as many bits as the rows of $\mathbf{A}$, i.e., K bits, and does not require subsequent entropy coding.

As discussed in [1], the modified quantizer enables efficient universal encoding of signals. Furthermore, this quantization method is also an embedding [19], satisfying with overwhelming probability on the measure of $\mathbf{A}$ and $\mathbf{w}$:

$$g\left(\|\mathbf{x}-\mathbf{y}\|_2\right) - \tau \leq d_H\left(f(\mathbf{x}), f(\mathbf{y})\right) \leq g\left(\|\mathbf{x}-\mathbf{y}\|_2\right) + \tau, \tag{2}$$

where $d_H(\cdot,\cdot)$ is the Hamming distance of the embedded signals and $g(d)$ is the map

$$g(d) = \frac{1}{2} - \sum_{i=0}^{+\infty} \frac{e^{-\left(\frac{\pi(2i+1)\sigma d}{\sqrt{2}\Delta}\right)^2}}{\left(\pi(i+1/2)\right)^2}, \tag{3}$$

which can be bounded using the bounds

$$g(d) \geq \frac{1}{2} - \frac{1}{2}e^{-\left(\frac{\pi\sigma d}{\sqrt{2}\Delta}\right)^2}, \quad g(d) \leq \frac{1}{2} - \frac{4}{\pi^2}e^{-\left(\frac{\pi\sigma d}{\sqrt{2}\Delta}\right)^2}, \quad g(d) \leq \sqrt{\frac{2}{\pi}}\frac{\sigma d}{\Delta}, \tag{4}$$

as shown in Fig. 2(b). The map is approximately linear for small d and becomes a constant equal to $1/2$ exponentially fast for large d, greater than a distance threshold D_0. The slope of the linear section and the distance threshold D_0 are determined by the embedding parameters, Δ and $\mathbf{A}$. In other words, the embedding ensures that the Hamming distance of the embedded signals is approximately proportional to the signals' ℓ_2 distance, as long as that ℓ_2 distance is smaller than D_0. Note that a piecewise linear function with slope $\sqrt{\frac{2}{\pi}}\frac{\sigma}{\Delta}$ until $d = D_0$ and slope equal to zero after that is a very good approximation to (3), in addition to being an upper bound.

The additive ambiguity τ in (2) scales as $\tau \propto 1/\sqrt{K}$, similar to the constant ϵ in the multiplicative $(1\pm\epsilon)$ factor in J-L embeddings. It should be noted, however, that universal embeddings use 1 bit per projection dimension, for a total rate of $R = K$. The trade-off between B and K under constant R exhibited by quantized J-L embeddings *does not exist* under the 1-bit universal embeddings. Still, there is a performance trade-off, controlled by the choice of Δ in (1), which we discuss in the next section.

Figure 2(c) demonstrates experimentally and provides intuition on how the embedding behaves for smaller (red) and larger (blue) Δ and for higher (left) and lower (right) bitrates. The figure plots the embedding (Hamming) distance as a function of the signal distance for randomly generated pairs of signals. The thickness of the curve is quantified by τ, whereas the slope of the upward sloping part is quantified by Δ.

Although not immediately relevant to this work, an information-theoretic argument guarantees that our embeddings can preserve the query's privacy [19].

III. Error Analysis of Distance Embeddings

A. General Error Analysis

To understand the ambiguities introduced by embeddings, we consider a general form of the distance guarantees provided by most embeddings. Specifically, consider an embedding $f : \mathcal{S} \to \mathcal{W}$ and distance metrics $d_{\mathcal{S}}(\cdot,\cdot)$ and $d_{\mathcal{W}}(\cdot,\cdot)$ in the signal space and the embedding space, respectively. This is a (g, ϵ, τ) embedding if, for all $\mathbf{s} \in \mathcal{S}$, it satisfies

$$(1-\epsilon)g\left(d_{\mathcal{S}}(\mathbf{x},\mathbf{y})\right) - \tau \leq d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right) \leq (1+\epsilon)g\left(d_{\mathcal{S}}(\mathbf{x},\mathbf{y})\right) + \tau, \tag{5}$$

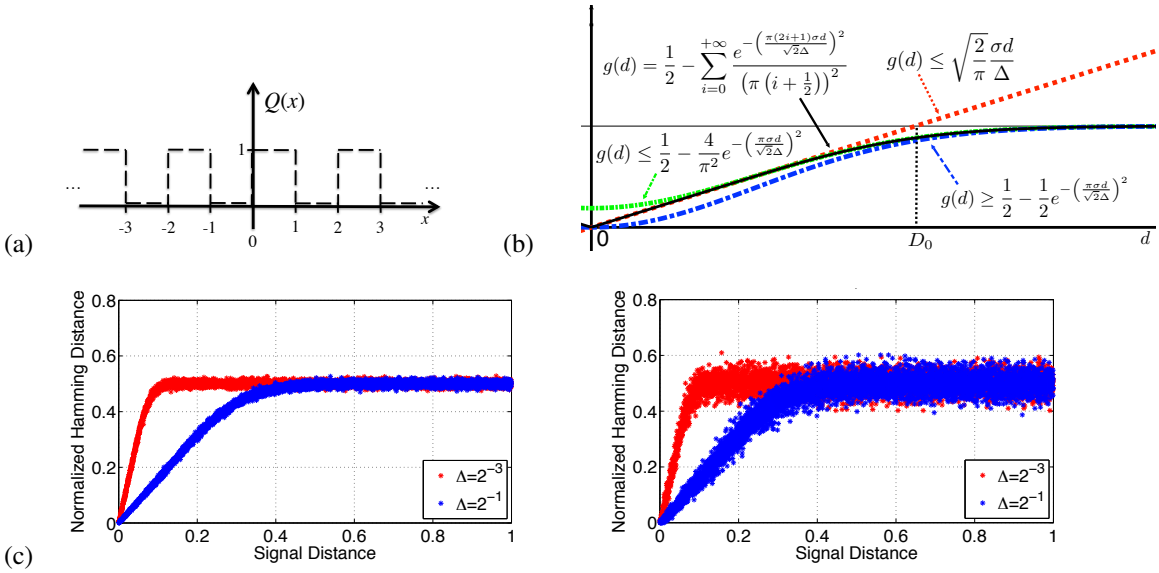

Fig. 2. (a) This non-monotonic quantization function $Q(\cdot)$ allows for universal rate-efficient scalar quantization. This function is equivalent to using a classical multibit scalar quantizer, and preserving only the least significant bit while discarding all other bits. (b) The embedding map $g(d)$ and its bounds produced by the quantization function (a). (c) Experimental verification of the embedding for small and large Δ in high (left) and low (right) bitrates.

where $g : \mathbb{R} \to \mathbb{R}$ is an invertible function mapping distances in $\mathcal{S}$ to distances in $\mathcal{W}$ and ϵ and τ quantify, respectively, the multiplicative and the additive ambiguity of the map.

To understand the performance of an embedding in distance computation we want to understand how well the embedding captures the distance. The main question is: given a distance $d_{\mathcal{W}}$ between two embedded signals in the embedding space $\mathcal{W}$, how confident are we about the corresponding distance between the original signals in the signal space $\mathcal{S}$? The function $g(\cdot)$ captures how distance is mapped and can be inverted to approximately determine the distance $d_{\mathcal{S}}$ in the signal space. On the other hand, the constants ϵ and τ capture the ambiguity in the opposite direction, i.e., the ambiguity in the embedding space given the distance in the signal space. Pictorially, taking Fig. 2(c) as an example, (5) characterizes the thickness of the curves taking a vertical slice of the plots, while we are now interested in the thickness revealed by taking a horizontal slice instead.

To capture the desired ambiguity, we can reformulate the embedding guarantees as

$$g^{-1}\left(\frac{d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right) - \tau}{(1+\epsilon)}\right) \le d_{\mathcal{S}}(\mathbf{x}, \mathbf{y}) \le g^{-1}\left(\frac{d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right) + \tau}{(1-\epsilon)}\right), \tag{6}$$

which for small ϵ and τ can be approximated using the Taylor expansion of $1/(1 \pm \epsilon)$:

$$g^{-1}\left(\left(d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right) - \tau\right)(1-\epsilon)\right) \lesssim d_{\mathcal{S}}(\mathbf{x}, \mathbf{y}) \lesssim g^{-1}\left(\left(d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right) + \tau\right)(1+\epsilon)\right), \tag{7}$$

Assuming that $g(\cdot)$ is differentiable, we can approximate the inequality using the Taylor expansion of $g^{-1}(\cdot)$ around $d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right)$ and the fact that $(g^{-1})'(x) = 1/g'(g^{-1}(x))$. Ignoring the second order term involving $\tau \cdot \epsilon$, and defining the signal distance estimate

$\widetilde{d}_{\mathcal{S}} = g^{-1}\left(d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right)\right)$ we obtain

$$\widetilde{d}_{\mathcal{S}} - \frac{\tau + \epsilon d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right)}{g'\left(\widetilde{d}_{\mathcal{S}}\right)} \lesssim d_{\mathcal{S}}(\mathbf{x}, \mathbf{y}) \lesssim \widetilde{d}_{\mathcal{S}} + \frac{\tau + \epsilon d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right)}{g'\left(\widetilde{d}_{\mathcal{S}}\right)}. \tag{8}$$

In other words, given the distance $d_{\mathcal{S}}$ between two signals in the signal space and using $\widetilde{d}_{\mathcal{S}}$ to denote the estimate of this distance, the ambiguity is less than

$$\left| d_{\mathcal{S}}(\mathbf{x}, \mathbf{y}) - \widetilde{d}_{\mathcal{S}} \right| \lesssim \frac{(\tau + \epsilon d_{\mathcal{W}}\left(f(\mathbf{x}), f(\mathbf{y})\right)}{g'\left(\widetilde{d}_{\mathcal{S}}\right)}. \tag{9}$$

Thus, ambiguity decreases by decreasing ϵ or τ, or by increasing the slope of the mapping.

B. Quantized J-L Embeddings

In quantized J-L embeddings, $g(d) = d$, which has constant slope equal to 1. Thus, the denominator in (9) is constant. To reduce the ambiguity a system designer should reduce the numerator as much as possible. To do so, as discussed in [4], the designer confronts the trade-off between the size of ϵ and τ. The former is controlled by the dimensionality of the projection, K, while the latter by the bit-rate per dimension, B. The greater K is, the smaller ϵ is. Similarly, the greater B is, the smaller τ is.

As we mention above, the total bit-rate of the embedding is equal to $R = KB$. In order to best use a given rate, the system designer should explore the trade-off between fewer projection dimensions at more bits per dimension and more projection dimensions at fewer bits per dimension. This trade-off is explored in detail in [4], where it is shown that, in the image retrieval application considered, the best performance is achieved using $B = 3$ or 4 bits per dimension and $K = R/3$ or $R/4$ dimensions, respectively. The performance of the two choices is virtually indistinguishable and significantly better than previous 1-bit approaches, using $B = 1$, $R = K$ [2], [3].

C. Universal Embeddings

In contrast to quantized J-L embeddings, universal embeddings use 1 bit per embedding dimension. Thus, the rate R also determines the dimensionality of the projection, $K = R$, as well as the constant τ in the embedding guarantees (2). Furthermore, there is no multiplicative term in the guarantees, i.e., $\epsilon = 0$. Thus, in the ambiguity analysis (9), the numerator is fully determined; the system designer can only control the denominator.

This does not mean that there are no design choices and trade-offs: the trade-off in these embedding is in the choice of the parameter Δ in (1). As discussed in the previous section and shown in Fig. 2(b), $g(\cdot)$ exhibits an approximately linear region, followed by a rapid flattening and an approximately flat region. The choice of Δ controls the slope of the linear region and, therefore, how soon the function reaches the flat region.

As mentioned earlier, the linear bound in (4) is a very good approximation of the upwards sloping linear region of $g(\cdot)$, which has slope $g'(d) \approx \sqrt{2/\pi}/\Delta$. By decreasing Δ, we can make that slope arbitrarily high, with a corresponding decrease of the ambiguity $\tau/g'(\widetilde{d}_{\mathcal{S}})$. However, this linear region does not extend for all d, but only until it reaches the point $d = D_0$ where $g(D_0) \approx 1/2$ and the flat region of $g(d)$ begins. As Δ becomes

smaller and the slope of the linear region increases, it reaches the flat region much faster, approximately when $D_0\sqrt{2/\pi}/\Delta = 1/2$, i.e., when $D_0 \approx \Delta\sqrt{\pi/8} \approx 0.6\Delta$.

Unfortunately, beyond that linear region, the slope $g'(d)$ becomes 0 exponentially fast. This implies that the ambiguity in (9) approaches infinity. Thus, if the embedding distance $d_{\mathcal{W}}$ is within $0.5 \pm \tau$, then it is impossible to know anything about $d_{\mathcal{S}}$ by inverting the mapping, other than $d_{\mathcal{S}} \gtrsim D_0$. This makes the trade-off in designing Δ clear. A smaller Δ reduces the ambiguity in the range of distances it preserves, but also reduces the range of distances it preserves. The system designer should design Δ such that the distances required in the application of the embedding are sufficiently preserved.

As an example, consider our motivating application: retrieval of nearest-neighbors from a database. When a query is executed, its embedding distance is computed with respect to all the entries in the database, embedded using the same parameters. For the query to be successful, there should be at least a few entries in the database will small embedding distance from the query. These entries are selected and returned. For the query to produce meaningful results, the embedding distance of those entries should represent quite accurately the signal distance between the query signal and the signals from the entries in the database. Furthermore, if the signals are all very distant from the query, the embedding distance should accurately reflect that fact, so that no signal is selected; in this case the embedding does not need to represent how distant each entry is.

In other words, the embedding only needs to represent distances up to a radius D, determined by the system designer, and to only *identify* distances further than D, without necessarily representing those distances. Thus, Δ should be designed to be as small as possible so the ambiguity in representing distances in the linear region is small, but not smaller than necessary to ensure that all distances of interest stay in the linear region of the embedding and not in the flat region with high ambiguity.

IV. Image Retrieval Using Universal Embeddings

A user wants to retrieve information about a query object by capturing its photograph and transmitting information extracted from the photograph to a database server. The server locates the object in the database that most closely matches the query image according to a predetermined distance criterion, and transmits the meta-data of that object back to the user. As we describe below, these tasks can be effectively accomplished by computing embeddings of features extracted from the query and database images.

A. Database Preparation

The server generates the embedding parameters—such as $\mathbf{A}$, $\mathbf{w}$, and Δ in the case of universal embeddings—according to the embedding specifications. 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, it runs a scale-invariant feature extraction algorithm on each image $\mathbf{I}_t$ to generate several feature vectors from each image. The number of features obtained from each image depends on parameters 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 $\{f(\mathbf{y}_1), \ldots, f(\mathbf{y}_L)\}$, where each $f(\mathbf{y}_i)$ is an R-bit quantized embedding of $\mathbf{y}_i$. As a final book-keeping step,

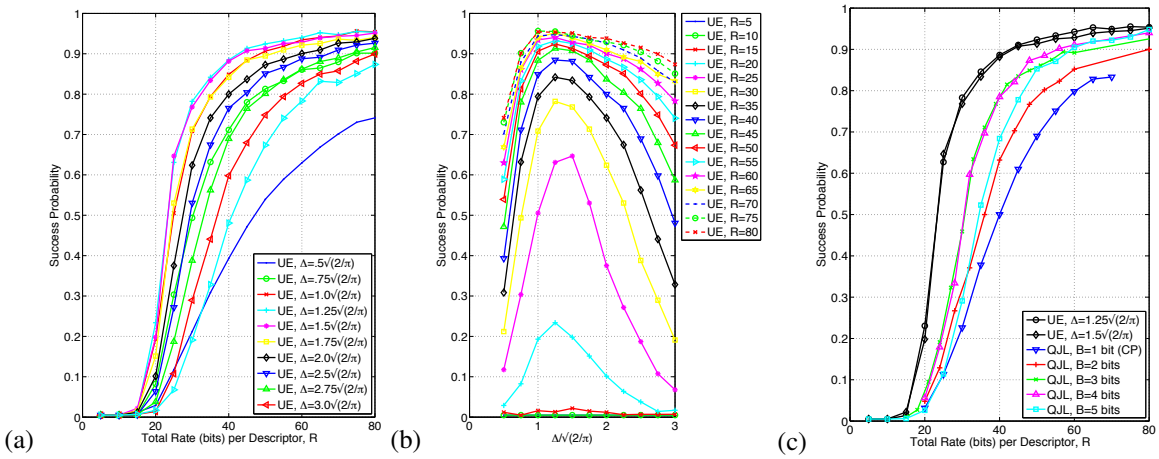

Fig. 3. Performance of universal embeddings (UE) in metadata retrieval. (a) Probability of correct retrieval as a function of the bitrate for a variety of Δ values. (b) Probability of correct retrieval as a function of Δ for a variety of bitrates. (c) Comparison of universal embeddings using $\Delta = 1.25\sqrt{2/\pi}$ and $1.5\sqrt{2.\pi}$ with quantized J-L methods (QJL). Universal embeddings significantly outperform the alternatives.

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 $f(\mathbf{y}_l)$ (or equivalently $\mathbf{y}_l$) was extracted.

B. Client Query

The client obtains the embedding parameters from the server, as a one-time software update or included as part of the client software installation. Once the client acquires the query image it executes the scale-invariant feature extraction algorithm to derive a set of features $\{\mathbf{x}_1, \ldots, \mathbf{x}_M\}$, where $\mathbf{x}_m$ is a descriptor corresponding to the m^{th} key (salient) point in the image. Using these M features and the embedding parameters, the client computes and transmits to the server the corresponding embeddings $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_M)\}$.

C. Approximate Nearest Neighbor Search and Meta-Data Retrieval

The server receives $\{f(\mathbf{x}_1), \ldots, f(\mathbf{x}_M)\}$ from the client. For each of the $f(\mathbf{x}_m)$ it computes the nearest neighbor among its database, i.e., among $\{f(\mathbf{y}_1), \ldots, f(\mathbf{y}_L)\}$. The result is M nearest neighbor pairs, one pair for each embedding $f(\mathbf{x}_m)$. Out of these M pairs, the server chooses the J pairs $\{f(\mathbf{x}_{(j)}), f(\mathbf{y}_{(j)})\}, j = 1, 2, ..., J$ that are closest in embedding distance (in our experiments, $J = 20$). 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.

V. Experimental Results

To validate our approach, we conducted metadata retrieval experiments using the ZuBuD database [15]. This public database contains 1005 images of 201 buildings in the city of Zurich. There are 5 images of each building taken from different viewpoints, all of size 640×480 pixels and compressed in PNG format. Our experimental setup is identical to [4]: One out of the 5 viewpoints of each building was randomly selected as

a query image, forming a test set of $s = 201$ images. The server's database comprises of the remaining 4 images of each building, for a total of $t = 804$ images. The query aims to identify which of the 201 possible buildings is depicted in each query image.

Our goal is to examine the performance of embeddings in preserving distances, not the performance of various feature selection methods. Thus, we extracted the widely adopted SIFT features [5] from each image and embedded them using quantized J-L embeddings or universal embeddings. Using the protocols described in Sec. IV we measured how many of the 201 query images produced the correct result, i.e., correctly identified the building depicted. We conducted our experiments in bitrates ranging from 0 to 80 bits per descriptor. Our results are averaged over 100 experiments with different realizations of $\mathbf{A}$ and $\mathbf{w}$, although the variability among individual runs was very small.

The first experiment tested the effect of Δ in the design of the embedding. We examined the range $\Delta = 0.5\sqrt{2/\pi}, 0.75\sqrt{2/\pi}, \ldots, 3\sqrt{2/\pi}$. The results are shown in Figs. 3(a) and (b). In Fig. 3(a) each curve plots the probability of correct metadata retrieval as a function of the bitrate used per descriptor, given a fixed Δ. The higher the probability of success, the better. Figure 3(b) presents another view on the same data: each curve plots the probability of correct retrieval given a fixed bitrate per descriptor as Δ varies.

The plots in Fig. 3(a) and (b) verify our expectations. As the bitrate increases, the performance improves. With respect to Δ, the behavior is more nuanced. For small Δ, the slope of $g(d)$ is high and the ambiguity in the linear region of $g(d)$ is low, as discussed in Sec. III-C. Thus, the embedding represents some distances very well. However, D_0 is small, i.e. it can only represent accurately a very small range of distances. Thus, for a large number of queries for which the closest matches are farther than D_0 the results returned are not meaningful. This type of error dominates the results when Δ is low. As Δ increases, more and more queries produce meaningful results and the error performance improves, even though the accuracy of the linear region of the embedding decreases. For larger Δ the reduced accuracy of the embedding starts dominating the error and the performance decreases again. The best performance is obtained for $\Delta = 1.25\sqrt{2/\pi}$, which corresponds to corresponding $D_0 = .625$.

We also compared the performance of our approach with existing methods based on quantized J-L embeddings. Figure 3(c) compares the performance of the two approaches The figure plots the probability of correct retrieval as a function of the bitrate per descriptor for each of the methods examined. As expected [4], multibit quantized J-L embeddings outperform 1-bit quantized J-L embeddings—known as "compact projections" (CP) [2], [3] and motivated by LSH approaches [14] in earlier literature. More important, universal embeddings—plotted in black circles and black diamonds, for $\Delta = 1.25\sqrt{2/\pi}$ and $1.5\sqrt{2.\pi}$ respectively—significantly outperform the other approaches. For example, to achieve a probability of correct retrieval of 80%, universal embeddings require approximately 8 fewer bits per descriptor, a 20% rate reduction. For 90% probability of correct retrieval, universal embeddings require 15 fewer bits per descriptor, a 25% rate reduction. Similarly, using only 40 bits per descriptor, universal embeddings achieve almost 90% success rate, versus almost 80% for the best alternative. The results are robust to Δ: for $\Delta \in [\sqrt{2/\pi}, 2\sqrt{2/\pi}]$, universal embeddings outperform all quantized J-L embeddings.

VI. Discussion and Conclusions

In summary, we have demonstrated that quantized embeddings are a powerful tool in encoding signals such that their pairwise distances are preserved. Our development

provides the tools necessary to understand the performance of such embeddings in this task, and to design them and use them according to the needs of the application at hand.

In the specific problem of identifying the nearest neighbors, only small distances need to be preserved by the encoding. In this case universal quantized embeddings outperform quantized Johnson-Lindenstrauss embeddings thanks to unequal preservation of distances. In particular, universal embeddings preserve distances very accurately up to a certain distance but not beyond that; quantized J-L embeddings preserve all distances equally, but not as accurately. In applications in which a larger range of distances should be preserved, we expect this advantage to diminish or disappear.

Of course, we have only scratched the surface of this very interesting topic. It is still an open question whether more efficient methods exist to encode distances between signals, as well as what the fundamental rate-distortion bounds are for this task.

REFERENCES

[1] P. T. Boufounos, "Universal rate-efficient scalar quantization," *IEEE Trans. Info. Theory*, vol. 58, no. 3, pp. 1861–1872, March 2012.

[2] 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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, San Francisco, CA, June 13–18 2010.

[3] C. Yeo, P. Ahammad, and K. Ramchandran, "Rate-efficient visual correspondences using random projections," in *Proc. IEEE International Conference on Image Processing (ICIP)*, San Diego, CA, October 12-15 2008.

[4] M. Li, S. Rane, and P. T. Boufounos, "Quantized embeddings of scale-invariant image features for mobile augmented reality," in *Proc. IEEE International Workshop on Multimedia Signal Processing (MMSP)*, Banff, Canada, Sept. 17–19 2012.

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

[6] 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, Jun. 2008.

[7] 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.

[8] A. Torralba, R. Fergus, and Y. Weiss, "Small codes and large image databases for recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Anchorage, AK, June 24-26 2008.

[9] Y. Weiss, A. Torralba, and R. Fergus, "Spectral hashing," in *Advances in Neural Information Processing Systems 21*, 2009, pp. 1753–1760.

[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*, vol. 34, no. 9, pp. 1704–1716, Sep. 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*, vol. 94, pp. 267–281, 2011.

[12] 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.

[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*, vol. 96, pp. 384–399, 2012.

[14] 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.

[15] 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/

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

[17] Y. Plan and R. Vershynin, "Dimension reduction by random hyperplane tessellations," *Arxiv preprint arXiv:1111.4452*, Nov. 2011.

[18] 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.

[19] 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.